University of Alberta

Modular data of finite groups

by

Jeremy David Macdonald ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**

Department of Mathematical and Statistical Sciences

Edmonton, Alberta
Fall 2006

Your file Votre référence
ISBN: 978-0-494-22308-6
Our file Notre référence
ISBN: 978-0-494-22308-6

University of Alberta

Library Release Form

Name of Author: Jeremy David Macdonald

Title of Thesis: Modular data of finite groups

Degree: Master of Science

Year this Degree Granted: 2006

Jeremy David Macdonald
10527-136 street
Edmonton, Alberta
Canada, T5N 2G1

Date: ____________

Abstract

The modular data associated with a finite group G is a representation of $\mathrm{SL}_2(\mathbb{Z})$, generated by matrices S and T, arising in conformal field theory and other contexts. A group's modular data determines many group-theoretic properties but it is unclear whether it determines the group. Under the naive definition of equivalence it does not determine the group, but we study a more restrictive definition under which groups of order less than 128 are distinguished. We make some remarks comparing modular data with 2-characters, giving an example of groups with equivalent 2-characters but inequivalent modular data. In the conformal field theory context, matrices commuting with S and T are of importance (modular invariants). We examine the algebra of such matrices, giving its dimension for the cyclic group $\mathbb{Z}_n$. Decomposing the representation into irreducibles is related to the study of this algebra and we give some results for $\mathbb{Z}_n$ and the dihedral group D_n, including the decomposition for $\mathbb{Z}_p$ (p a prime).

Acknowledgements

I thank my supervisor, Dr. Terry Gannon, for his guidance, advice, and patience.

I thank my family who supported me despite not understanding what it is I do.

I would like to acknowledge financial support from NSERC and the Department of Mathematical and Statistical Sciences, University of Alberta.

Thanks to Andrew and Quan for providing a stimulating work environment.

Contents

List of Symbols

General symbols

G, H	finite groups
$H \leq G$	H is a subgroup of G
e	group identity
$\langle x, y \rangle$	group generated by x and y
$K(g)$	conjugacy class of g
$C(g)$	centralizer in g
$Z(G)$	centre of group G
C_n	cyclic group of order n
D_n	dihedral group of order $2n$
Q_{2n}	quaternion group of order $4n$
S_n	symmetric group on n symbols
B_n	braid group on n strings
$\mathbb{Z}_n$	$\mathbb{Z}/n\mathbb{Z}$ (integers modulo n)
$x \equiv_n y$	x and y are equivalent modulo n
$M_m(B)$	algebra of $m \times m$ matrices over ring B
$\mathrm{SL}_2(\mathbb{Z})$	$\{A \in M_2(\mathbb{Z}) \mid \det A = 1\}$
$\mathrm{SL}_2(\mathbb{Z}_m)$	$\{A \in M_2(\mathbb{Z}_m) \mid \det A = 1\}$
ξ_n	primitive n^{th} root of unity $e^{\frac{2\pi i}{n}}$
$< \chi, \psi >$	inner product of characters
s, t	generators of $\mathrm{SL}_2(\mathbb{Z})$
$\mathbb{H}$	upper half-plane of $\mathbb{C}$
$\bar{z}$	complex-conjugate of $z \in \mathbb{C}$

Symbols relating to modular data

$R = R(G)$	a set of representatives of the conjugacy classes of G
R_a	a set of representatives of the conjugacy classes of $C_G(a)$
$C^0(G_{comm})$	$\mathbb{C}$-valued functions on $G \times G$ constant under simultaneous conjugation
$\mathcal{P}$	permutation basis of $C^0(G_{comm})$
$[\![a, b]\!]$	element of $\mathcal{P}$
$\mathcal{C}$	character basis of $C^0(G_{comm})$
ch^a_χ	element of $\mathcal{C}$
ρ_G	modular data (representation of $\mathrm{SL}_2(\mathbb{Z})$ defined by G)
$S^G = S$	$\rho_G(s)$
$T^G = T$	$\rho_G(t)$
$\mathrm{Sym}(\mathcal{C})$	group of valid permutations of basis $\mathcal{C}$

Chapter 1

Introduction and background

1.1 Introduction

Associated with every finite group G is a representation ρ_G of the modular group $\mathrm{SL}_2(\mathbb{Z})$. $\mathrm{SL}_2(\mathbb{Z})$ is important due to its relation with the moduli space of tori (discussed in §1.4). The representation arises in several contexts, most importantly in conformal field theory (an extremely symmetrical quantum field theory that has applications in string theory). We will review these contexts in §2.3. In the conformal field theory context there is a 'preferred' basis $\mathcal{C}$ for the representation space, so ρ_G is a matrix representation, in particular defined by matrices S and T (the images under ρ_G of generators of $\mathrm{SL}_2(\mathbb{Z})$). In this context ρ_G is called the *modular data* associated with G. There is also a basis $\mathcal{P}$ in which ρ_G is a permutation representation. We will define ρ_G and these bases in Chapter 2.

A major question is to what extent S and T determine G (i.e. how strong of a group invariant is it). This depends on the definition of equivalence of S and T used. Under the naive definition, it seems that S and T only determine G up to order 15 ([Cun05]). We propose a more restrictive definition and prove, computationally, that S and T determine G for groups of order less than 128 (Chapter 4). This is one of the most important original results of the thesis. In Chapter 5 we review two important *complete invariants* of finite groups — the group determinant and k-characters — and make some comparisons with modular data. We include a new result giving groups with equivalent *2-characters* but inequivalent modular data.

In the conformal field theory context, the *centralizer algebra* of ρ_G plays a key role in determining the possible *modular invariants* (an important quantity in the conformal field theory, see Chapter 3). For modular data arising from finite groups, neither the centralizer algebra nor the modular invariants have been well-explored (modular data arises in other ways in conformal field theory, for example from affine algebras, and has been studied more carefully). In Chapter 3 we give some new results on the centralizer algebra including its dimension for cyclic groups. We give some results on decomposing ρ_G into irreducible representations (ρ_G is in fact a representation of a finite quotient of $\mathrm{SL}_2(\mathbb{Z})$) which help to understand the centralizer algebra.

The remainder of this chapter consists of background material we will be needing later on. This is all standard material. The reader planning to skip ahead to Chapter 2 is advised to pause at §1.4 which deals with $\mathrm{SL}_2(\mathbb{Z})$.

1.2 Group theory background

We start with some group theory basics and fix some notation. A good reference for elementary group theory is [DF99].

Let G and H to denote finite groups throughout, with e denoting the group identity. Let $g,\ h \in G$.

- $C_G(g) = \{x \in G \mid xg = gx\}$ is the *centralizer of* g and is a subgroup. When the group G is clear we will write $C(g)$.
- $Z(G) = \{g \in G \mid gx = xg \quad \text{for all } x \in G\}$ is the *center of* G and is a subgroup.
- If there exists $x \in G$ such that $xgx^{-1} = h$ then g and h are said to be *conjugate (in G)*. Conjugacy is denoted $g \sim_G h$, or simply $g \sim h$.
- $K_G(g) = \{xgx^{-1} \mid x \in G\}$ is the *conjugacy class of* g. When the group is clear we will write $K(a)$.
- The *order* of G is its cardinality as a set and is denoted $|G|$.
- The *order* of g is the least positive integer m such that $g^m = e$ and is denoted $|g|$.
- The *exponent* of G is the least positive integer m such that $g^m = e$ for all $g \in G$. Equivalently, it is the least common multiple of the orders of all the elements of G,
$$\mathrm{Exponent}(G) = \mathrm{lcm}\{|g| \mid g \in G\}$$

With regard to conjugacy classes, we remark that

- Conjugacy is an equivalence relation, and G is partitioned into conjugacy classes.
- Conjugate elements have the same order. This follows from the equation $(xgx^{-1})^l = xg^lx^{-1}$.

1.2.1 Group actions

For a set Ω, a *(right) group action* of G on Ω is a map $\Omega \times G \longrightarrow \Omega$, $(\omega, g) \longmapsto \omega.g$ satisfying

(a) $\omega.(gh) = (\omega.g).h$ for all $g,\ h \in G$, $\omega \in \Omega$.

(b) $\omega.e = \omega$ for all $\omega \in \Omega$

We say that *G acts on Ω*. A *left* group action is define analogously. The *orbit* of ω is $\mathrm{Orb}(\omega) = \{\omega.g \mid g \in G\}$. The set of all orbits is denoted Ω/G and forms a partition of Ω. The *stabilizer* of ω is $\mathrm{Stab}_G(\omega) = \{g \in G \mid \omega.g = \omega\}$ and is a subgroup of G. Every group acts on itself by conjugation, the action being $g.x = gxg^{-1}$ for $x, g \in G$ (this is left group action, defining $x.g = g^{-1}xg$ gives a right group action). We give two important lemmas about group actions. For proofs, see Chapter 4 Proposition 2 of [DF99] and Theorem 2.2 of [Cam99].

Lemma 1.1 *Let G act on Ω, $g \in G$, $\omega \in \Omega$. Then*

$$|\mathrm{Orb}(\omega)| = \frac{|G|}{|\mathrm{Stab}_G(\omega)|}$$

In particular, when $\Omega = G$ and the action is conjugation, the formula becomes

$$|K(g)| = \frac{|G|}{|C(g)|}$$

Lemma 1.2 ('Burnside's Lemma') *Let G act on Ω and define* $\mathrm{Fix}(g) = \{\omega \in \Omega \mid \omega.g = \omega\}$. *Then number of orbits of the action is* $\frac{1}{|G|}\sum_{g \in G} |\mathrm{Fix}(g)|$.

Though the lemma is often called Burnside's Lemma, it is not due to Burnside (it was known to Frobenius earlier).

1.2.2 Generators and relations

One way of describing a group is with *generators* and *relations*. To do this formally, one forms a quotient of a free group. This construction can be found in [DF99], but we will content ourselves with an example. The dihedral group D_n of order $2n$ is described using generators and relations as

$$D_n = \langle x, y \mid y^n = e,\ x^2 = e,\ yx = xy^{-1} \rangle$$

The generators of the group are the symbols x and y and the relations are $y^n = e$, $x^2 = e$, and $yx = xy^{-1}$. A *word* in the symbols x,y,x^{-1},y^{-1} is just a sequence composed of these symbols, for example $xyx^{-1}yy^{-1}$ or $yxxxxx^{-1}$. These words form the group elements, in addition to the group identity e (which can be represented by the 'empty' word consisting of no symbols). Two words are multiplied by concatenation, e.g. xy multiplied by yx is $xyyx$. We write $xx\cdots x$ (n times) as x^n, and $xx^{-1} = x^{-1}x = e$. The relations specify further simplifications that we can apply to words. For example, the element $yxxxxx^{-1}$ can be simplified to yx^3, then to $yxe = yx$ using the relation $x^2 = e$, then to xy^{-1} using the relation $yx = xy^{-1}$ (though this last step is arguably not a 'simplification').

For finite groups, we can often find a canonical representation of each element. In the dihedral group above, the relation $yx = xy^{-1}$ allows us to interchange x and y (note that the relation also implies $xy = y^{-1}x$), meaning we can write any element in the form x^iy^j for some $i, j \in \mathbb{Z}$. Since $x^2 = e$ and $y^n = e$ we need only use $i \in \{0, 1\}$ and $j \in \{0, 1, \ldots, n-1\}$. These elements are all distinct (a proof is required), hence the group has order $2n$. In general, determining whether two words are equal (represent the same group element) is an *undecidable* problem.

1.3 Representation and character theory of finite groups

In this section we present a brief introduction to representation theory and character theory, including several results that we will need later. A good introduction to character theory is given in [Gro97], and the presentation here is based on that source. Proofs can be found there.

1.3.1 Representations of finite groups

Let V be an m-dimensional vector space over $\mathbb{C}$. A *representation of G on V* is a homomorphism $\varphi : G \longrightarrow \mathrm{GL}(V)$, where $\mathrm{GL}(V)$ is the group of nonsingular linear transformations from V to V. If we fix a basis of V, we get $\mathrm{GL}(V) \cong \mathrm{GL}(m, \mathbb{C})$ (the group of $m \times m$ invertible matrices over $\mathbb{C}$) and $\varphi : G \longrightarrow \mathrm{GL}(m, \mathbb{C})$ is called a *matrix representation*. We call m the *dimension* or *degree* of the representation.

Since $\varphi(g)$ is a linear transformation on V, we write $\varphi(g)v$ for the image of $v \in V$ under $\varphi(g)$. Representations over fields other than $\mathbb{C}$ are defined the same way, but we will only deal with $\mathbb{C}$-representations.

A subspace $U \subset V$ is called *φ-invariant* if $\varphi(g)U \subset U$ for all $g \in G$. If any non-zero φ-invariant subspaces exist then φ is called *reducible*, otherwise φ is *irreducible*. Two representations $\varphi : G \longrightarrow \mathrm{GL}(V)$ and $\phi : G \longrightarrow \mathrm{GL}(W)$ are *equivalent* if there exists a vector space isomorphism $L : V \longrightarrow W$ such that $L\varphi(g) = \phi(g)L$ for all $g \in G$. Write $\varphi \cong \phi$ for equivalent representations.

Define the *direct sum* $\varphi \oplus \phi$ of representations φ and ϕ on the vector space $V \oplus W$ by $(\varphi \oplus \phi)(g)(v, w) = (\varphi(g)v, \phi(g)w)$. When φ and ϕ are matrix representations, the direct sum is the block-diagonal matrix

$$(\varphi \oplus \phi)(g) = \begin{pmatrix} \varphi(g) & 0 \\ 0 & \phi(g) \end{pmatrix}$$

A representation is *completely reducible* if it is equivalent to a direct sum of irreducible representations. Define the *tensor product* $\varphi \otimes \phi$ of of φ and ϕ on $V \otimes W$ by $(\varphi \otimes \phi)(g)(v \otimes w) = (\varphi(g)v) \otimes (\phi(g)w)$. For matrix representations, $(\varphi \otimes \phi)(g)$ is the (Kronecker) tensor product $\varphi(g) \otimes \phi(g)$. We are interested in representations of finite groups.

Theorem 1.3 *Let k be the number of conjugacy classes of the finite group G. Then there are exactly k inequivalent irreducible $\mathbb{C}$-representations of G.*

This is NOT true for infinite groups. For example, the integers have uncountably many inequivalent irreducible representations. Take any $0 \neq z \in \mathbb{C}$. Then $\varphi_z(n) = z^n$ forms a family of inequivalent irreducible representations of $\mathbb{Z}$.

Every group has a 1-dimensional 'trivial representation', $1 : G \longrightarrow \mathrm{GL}(1, \mathbb{C})$, $1(g) = 1$. The trivial representation is one of the irreducible representations of G. For finite groups, we have the following important theorem:

Theorem 1.4 *Every $\mathbb{C}$-representation ϕ of a finite group G is completely reducible. If $\varphi_1, \varphi_2, \ldots, \varphi_k$ are the inequivalent irreducible representations of G, then there exist $m_1, m_2, \ldots, m_k \in \mathbb{Z}_{\geq 0}$ such that*

$$\phi \cong \bigoplus_{i=1}^{k} m_i \varphi_i$$

where $m_i\varphi_i = \oplus_{j=1}^{m_i} \varphi_i$.

Again, this is not the case for infinite groups. The $\mathbb{Z}$ representation $n \longmapsto \left(\begin{smallmatrix}1 & n\\ 0 & 1\end{smallmatrix}\right)$ is not completely reducible. Two useful results which we will use often are *Schur's Lemma* and one of its corollaries:

Lemma 1.5 (Schur's Lemma) *Let φ and π be irreducible $\mathbb{C}$-representations of G on the spaces V and W (respectively). Suppose $L : V \longrightarrow W$ is a linear transformation such that $L\varphi(g) = \pi(g)L$ for all $g \in G$. Then either $L = 0$ or L is an isomorphism (and representations are equivalent). Further, if $V = W$ then L is a scalar map.*

Corollary 1.6 *Let φ be an irreducible representation of G and $z \in Z(G)$. Then*

$$\varphi(z) = \xi I$$

where ξ is an $|z|^{th}$ root of unity and I is the identity matrix of size $\deg \varphi$.

Proof. For every $g \in G$ we have $zg = gz$ so $\varphi(zg) = \varphi(gz)$ and

$$\varphi(z)\varphi(g) = \varphi(g)\varphi(z)$$

Then by Schur's Lemma, $\varphi(z) = \xi I$ for some $\xi \in \mathbb{C}$. Since $x^{|x|} = e$ we have $I = \varphi(e) = \varphi(x)^{|x|} = \xi^{|x|} I$ hence $\xi^{|x|} = 1$.

□

1.3.2 Group actions on finite sets are permutation representations

A *permutation representation* of G is a matrix representation φ where $\varphi(g)$ is a permutation matrix (a matrix with exactly one 1 is each row and column, and 0 elsewhere). A permutation representation is the same as an action of G on a finite set. Suppose G acts on the finite set $\Omega = \{\omega_1, \omega_2, \dots, \omega_m\}$. Let Ω form a basis for $\mathbb{C}^m$. Then φ defined by

$$\varphi(g)\left(\sum_{i=1}^{m} c_i\omega_i\right) = \sum_{i=1}^{m} c_i\omega_i.g$$

is a permutation representation of dimension m. Indeed, writing φ as a matrix with respect to the basis Ω, we get

$$\varphi(g)_{i,j} = \begin{cases} 1 & \text{if } \omega_j.g = \omega_i \\ 0 & \text{otherwise} \end{cases}$$

Conversely, given a permutation representation φ on vector space V, let Ω be the basis in which $\varphi(g)$ is a permutation matrix and let G act on Ω by $\omega.g = \varphi(g)\omega$.

1.3.3 Character theory

Let φ be a (matrix) representation of G. Define the *character* χ of φ as the trace of φ:

$$\begin{aligned} \chi : G &\longrightarrow \mathbb{C} \\ g &\longmapsto \operatorname{tr}(\varphi(g)) \end{aligned}$$

Trace is invariant under change-of-basis, so the character is independent of the choice of basis. Equivalent representations have equal characters. Characters are *class functions* on G, meaning they are constant on conjugacy classes. Indeed,

$$\chi(hgh^{-1}) = \operatorname{tr}(\varphi(h)\varphi(g)\varphi(h)^{-1}) = \operatorname{tr}(\varphi(g)) = \chi(g)$$

Being class functions, we may write $\chi(K(g))$ instead of $\chi(g)$, where $K(g)$ is the conjugacy class of g. A character is called *irreducible* if the corresponding representation is irreducible. Since finite G has k (number of conjugacy classes) irreducible representations, it also has k irreducible characters. Let $\operatorname{Irr}(G)$ denote the set of irreducible characters of G.

Theorem 1.7 *Let $CF(G) = \{f : G \longrightarrow \mathbb{C} \mid f(g) = f(hgh^{-1})$ for all $g, h \in G\}$ be the $\mathbb{C}$-vector space of class functions on G. Then the set of irreducible characters of G forms a basis of $CF(G)$. In particular, the irreducible characters are linearly independent and $CF(G)$ has dimension k.*

The trivial representation is always irreducible, so G always has a trivial character $1(g) = 1$. Note that 1-dimensional representations are equal to their characters. Since $\chi(e)$ is the trace of the identity matrix, we have that $\chi(e)$ is the dimension m of the corresponding representation. We call m the *degree* of χ. The next theorem relates these degrees to the order of G, and helps in finding the irreducible characters.

Theorem 1.8 *Let $\chi_1, \chi_2, \ldots, \chi_k$ be the irreducible inequivalent characters of G. Then*

$$\sum_{i=1}^{k} \chi_i(e)^2 = |G|$$

Though characters are defined as functions into $\mathbb{C}$, the character values actually lie in a subring of $\mathbb{C}$:

Proposition 1.9 *Let χ be a character of G. Then for all $g \in G$, $\chi(g) \in \mathbb{Z}[\xi_m]$, where $m = \text{Exponent}(G)$ and ξ_m is the primitive m^{th} root of unity $\exp(2\pi i/m)$. Further, $\chi(g^{-1}) = \overline{\chi(g)}$, where bar denotes complex conjugation.*

Define a Hermitian form on $CF(G)$, which is extremely useful in character theory:

$$< \chi, \psi >= \frac{1}{|G|} \sum_{g \in G} \chi(g)\overline{\psi(g)}$$

Several important theorems can be stated or proved using this form.

Theorem 1.10 *Let ϕ be a representation of finite group G with decomposition into irreducibles $\varphi = \oplus_{i=1}^{k} m_i \varphi_i$. Let $\text{Irr}(G) = \{\chi_1, \chi_2, \cdots, \chi_k\}$, with χ_i being the character of φ_i. Then the character χ of ϕ decomposes as*

$$\chi = \sum_{i=1}^{k} m_i \chi_i$$

Further, the multiplicities are given by $m_i =< \chi, \chi_i >$.

Theorem 1.11 (First Orthogonality Relation) *Let χ_i, $\chi_j \in \text{Irr}(G)$. Then $< \chi_i, \chi_j >= \delta_{ij}$.*

Theorem 1.12 *Let $\chi \in \text{Irr}(G)$. Then χ is irreducible if and only if $< \chi, \chi >= 1$.*

Theorem 1.13 *Let φ_i be the irreducible representations of G and ϕ_j the irreducible representations of H. Then the irreducible representations of $G \times H$ are the $\varphi_i \otimes \phi_j$.*

Proof. First we show that the $\varphi_i \otimes \phi_j$ are irreducible. Let χ_i be the character of φ_i and ψ_j the character of ϕ_j. Then the character of $\varphi_i \otimes \phi_j$ is the product $\chi_i \cdot \psi_j$ (this is easy to see writing φ_i and ϕ_j as matrix representations), and we apply Theorem 1.12:

$$\begin{aligned}
< \chi_i \cdot \psi_j, \chi_i \cdot \psi_j > &= \frac{1}{|G \times H|} \sum_{(g,h) \in G \times H} \chi_i(g)\psi_j(h)\overline{\chi_i(g)\psi_j(h)} \\
&= \left(\frac{1}{|G|} \sum_{g \in G} \chi_i(g)\overline{\chi_i(g)} \right) \left(\frac{1}{|H|} \sum_{h \in H} \psi_j(h)\overline{\psi_j(h)} \right) \\
&= < \chi_i, \chi_i >< \psi_j, \psi_j > \\
&= 1
\end{aligned}$$

hence by Theorem 1.12 the corresponding representation is irreducible.

To show that these are all of the irreducible representations, we apply Theorem 1.8:

$$\sum_{i,j} \chi_i(e)^2 \psi_j(e)^2 = \left(\sum_i \chi_i(e)^2\right)\left(\sum_j \psi_j(e)^2\right) = |G||H| = |G \times H|$$

hence there are no other irreducible characters hence no other irreducible representations.

□

Theorem 1.14 (Second Orthogonality Relation) *Let $a_1, a_2, \ldots, a_k$ be representatives of the conjugacy classes of G. Then*

$$\sum_{\chi \in \mathrm{Irr}(G)} \chi(a_i)\overline{\chi(a_j)} = \delta_{ij}|C_G(a_i)|$$

1.3.4 Induced characters

Let $H \leq G$ and φ an m-dimensional matrix representation of H. We can get a representation of G from φ. Let $x_1H, x_2H, \ldots, x_kH$ be the left cosets of H in G. Define $\varphi_H^G : G \longrightarrow \mathrm{GL}(mk, \mathbb{C})$ by letting $\varphi_H^G(g)$ be the block matrix whose i, j block is

$$(\varphi_H^G)_{ij} = \varphi(x_i^{-1}gx_j)$$

where $\varphi(x_i^{-1}gx_j)$ is the $m \times m$ zero matrix whenever $x_i^{-1}gx_j \notin H$.

Proposition 1.15 *The function φ_H^G above is a representation of G, called the* induced representation. *Its character χ_H^G is called the* induced character *and is given by*

$$\chi_H^G(g) = \frac{1}{|H|} \sum_{\substack{y \in G \\ y^{-1}gy \in H}} \chi(y^{-1}gy)$$

Remark 1.16 *We need not fix a basis in order to define the induced character. It always exists and is unique up to equivalence. The restriction of χ_H^G to H is in general not equal to χ.*

1.3.5 Character tables

Let $K_1, K_2, \ldots, K_k$ be the conjugacy classes of G and $\chi_1, \chi_2, \ldots, \chi_k$ its irreducible characters. The *character table* of G is the array with columns indexed by the conjugacy classes and rows by the irreducible characters and whose (χ_i, K_j) entry is $\chi_i(K_j)$. Neither the rows nor the columns are ordered in any particular way, though one often writes the class of e first, the trivial character first, and the characters in order of increasing degree. See below for examples of character tables. Note that the number of irreducible characters is the same as the number of conjugacy classes, so the character table is square.

Let G have k conjugacy classes and irreducible characters as above, and suppose H also has k conjugacy classes. Then we say that G and H have *isomorphic character tables* if there exist bijections π, σ between the irreducible characters and conjugacy classes (respectively) of G and H such that $\chi_i(K_j) = (\pi\chi_i)(\sigma K_j)$ for all i, j. Permuting the rows of H's character table by π and its columns by σ produces G's character table. Non-isomorphic groups may have isomorphic character tables (see for example D_{2n} and Q_{2n} when n is even, described in the next section).

Character table of dihedral and quaternion groups

As an example, we will write down the character tables of the dihedral and quaternion groups. We will these character tables later.

The dihedral group of order $2n$ is given by the presentation $D_n = \langle x, y \mid x^2 = y^n = e,\ xy = y^{-1}x\rangle$. The elements can be listed as $x^i y^j$ where $i \in \{0,1\}$, $0 \le j < n$. Results are slightly different for the cases when n is even and when n is odd. When n is even, the conjugacy classes are

$$\begin{aligned}
K(e) &= \{e\} \\
K(x) &= \{xy^{2i} \mid 0 \le i \le \tfrac{n}{2} - 1\} \\
K(xy) &= \{xy^{2i+1} \mid 0 \le i \le \tfrac{n}{2} - 1\} \\
K(y^i) &= \{y^i, y^{n-i}\},\ 1 \le i \le \tfrac{n}{2} - 1 \\
K(y^{\frac{n}{2}}) &= \{y^{\frac{n}{2}}\}
\end{aligned}$$

and the character table is

D_n	$K(e)$	$K(x)$	$K(xy)$	$K(y^{\frac{n}{2}})$	$K(y^i)$
1	1	1	1	1	1
ψ_1	1	-1	-1	1	1
ψ_2	1	1	-1	$(-1)^{\frac{n}{2}}$	$(-1)^i$
ψ_3	1	-1	1	$(-1)^{\frac{n}{2}}$	$(-1)^i$
χ_j	2	0	0	$2(-1)^j$	$2\cos\left(\frac{2\pi ij}{n}\right)$

where $1 \leq i, j \leq \frac{n}{2} - 1$. For n odd, the conjugacy classes are

$$\begin{aligned} K(e) &= \{e\} \\ K(x) &= \{xy^i \mid 0 \leq i < n\} \\ K(y^i) &= \{y^i, y^{n-i}\},\ 1 \leq i \leq \frac{n-1}{2} \end{aligned}$$

and the character table is

D_n	$K(e)$	$K(x)$	$K(y^i)$
1	1	1	1
ψ_1	1	-1	1
χ_j	2	0	$2\cos\left(\frac{2\pi ij}{n}\right)$

where $1 \leq i, j \leq (n-1)/2$.

The (generalized) quaternion group of order $4n$ has the presentation $Q_{2n} = \langle x, y \mid y^{2n} = e,\ x^2 = y^n,\ yx = xy^{-1}\rangle$. Its elements can be listed as $x^i y^j$ where $i \in \{0, 1\}$ and $0 \leq j < 2n$. Conjugacy classes are given by

$$\begin{aligned} K(e) &= \{e\} \\ K(x) &= \{xy^{2i} \mid 0 \leq i \leq n-1\} \\ K(xy) &= \{xy^{2i+1} \mid 0 \leq i \leq n-1\} \\ K(y^i) &= \{y^i, y^{2n-i}\},\ 1 \leq i \leq n-1 \\ K(y^n) &= \{y^n\} \end{aligned}$$

The character table is slightly different for n odd and n even. Let $\iota = 1$ when n is even and $\iota = \boldsymbol{i}$ ($\boldsymbol{i}^2 = -1$) when n is odd. Then the character table is

Q_{2n}	$K(e)$	$K(x)$	$K(xy)$	$K(y^n)$	$K(y^j)$
1	1	1	1	1	1
ψ_1	1	-1	-1	1	1
ψ_2	1	ι	$-\iota$	$(-1)^n$	$(-1)^j$
ψ_3	1	$-\iota$	ι	$(-1)^n$	$(-1)^j$
χ_i	2	0	0	$2(-1)^i$	$2\cos\left(\frac{\pi ij}{n}\right)$

where $1 \leq i, j \leq n - 1$. Notice the similarity with the character table of D_n. In particular, when n is even Q_{2n} and D_{2n} have the same character tables.

1.4 The modular group $\mathrm{SL}_2(\mathbb{Z})$

The group $\mathrm{SL}_2(\mathbf{Z})$ is called the *modular group* and consists of all 2×2 integer matrices with determinant 1,

$$\mathrm{SL}_2(\mathbf{Z}) = \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in M_2(\mathbf{Z}) \mid ad - cb = 1 \right\}$$

Modular data is a representation of $\mathrm{SL}_2(\mathbf{Z})$ so we will be interested in it throughout. Of particular importance is that fact that $\mathrm{SL}_2(\mathbf{Z})$ is generated by two elements:

Proposition 1.17 *$\mathrm{SL}_2(\mathbf{Z})$ is generated by the elements $s = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ and $t = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$.*

Proof. Our proof is based on one given in [Apo76]. First we remark that $t^n = \left(\begin{smallmatrix} 1 & n \\ 0 & 1 \end{smallmatrix}\right)$ and $s^2 = -\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$. Let $A = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbf{Z})$. Assume $c \geq 0$ and proceed by induction on c. If $c = 0$, then since $\det(A) = 1$ we have $ad = 1$, so $a = d = 1$ or $a = d = -1$. In the first case, $A = \left(\begin{smallmatrix} 1 & b \\ 0 & 1 \end{smallmatrix}\right) = t^b$ and in the second case $A = -\left(\begin{smallmatrix} 1 & -b \\ 0 & 1 \end{smallmatrix}\right) = s^2 t^{-b}$. If $c = 1$, then $1 = \det(A) = ad - b$ so

$$A = \begin{pmatrix} a & ad-1 \\ 1 & d \end{pmatrix} = \begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & d \\ 0 & 1 \end{pmatrix} = t^a s t^d$$

Now assume $c > 1$ and that every matrix in $\mathrm{SL}_2(\mathbf{Z})$ with lower-left entry non-negative and less than c is generated by s and t. Since $ad - bc = 1$, we know $\gcd(c, d) = 1$ and in particular $d \neq 0$ and $d \neq c$ (remember $c > 1$). Then division of d by c gives

$$d = cq + r \quad 0 < r < c$$

and we have

$$At^{-q}s = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} 1 & -q \\ 0 & 1 \end{pmatrix} s = \begin{pmatrix} a & -aq+b \\ c & r \end{pmatrix} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} -aq+b & -a \\ r & -c \end{pmatrix}$$

Since $r < c$ we get by the induction assumption that $At^{-q}s$ is generated by s and t hence $A = (At^{-q}s)s^{-1}t^q$ is generated by s and t, completing the induction.

To complete the proof, if $c < 0$ then $As^2 = -A$ has lower-left entry non-negative hence is generated by s and t, so $A = (-A)s^{-2}$ is generated by s and t.

□

Remark 1.18 *In fact, every element of $\mathrm{SL}_2(\mathbf{Z})$ can be expressed using only positive powers of s and t since $s^4 = 1$ so $s^{-1} = s^3$ and $t^{-1} = \left(\begin{smallmatrix} 1 & -1 \\ 0 & 1 \end{smallmatrix}\right) = ststs^3$.*

Proposition 1.19 *A presentation for $\mathrm{SL}_2(\mathbf{Z})$ is $< s, t \mid s^4 = 1,\ s^2 = (st)^3 >$.*

Proof. Showing that $s = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ and $t = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$ satisfy the relations is easy. Showing that no that no other relations are needed is more difficult.

□

For a positive integer m define a subgroup $\Gamma(m) < \mathrm{SL}_2(\mathbb{Z})$ as

$$\Gamma(m) = \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathrm{SL}_2(\mathbb{Z}) \mid \begin{pmatrix} a & b \\ c & d \end{pmatrix} \equiv \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \pmod{m} \right\}$$

Lemma 1.20 *$\Gamma(m)$ is a normal subgroup of* $\mathrm{SL}_2(\mathbb{Z})$ *and* $\mathrm{SL}_2(\mathbb{Z})/\Gamma(m) \cong \mathrm{SL}_2(\mathbb{Z}_m)$.

Proof. One uses the obvious homomorphism $\varphi : \mathrm{SL}_2(\mathbb{Z}) \longrightarrow \mathrm{SL}_2(\mathbb{Z}_m)$, $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \overset{\varphi}{\longmapsto} \left(\begin{smallmatrix} (a \bmod m) & (b \bmod m) \\ (c \bmod m) & (d \bmod m) \end{smallmatrix}\right)$ and shows that $\ker(\varphi) = \Gamma(m)$ and $\mathrm{Im}(\varphi) = \mathrm{SL}_2(\mathbb{Z}_m)$ so the lemma follows from the 'First Isomorphism Theorem'. The only non-trivial step is showing $\mathrm{Im}(\varphi) = \mathrm{SL}_2(\mathbb{Z}_m)$. A proof is given in §6.1 of [Lan87].

□

$\mathrm{SL}_2(\mathbb{Z})$ acts the complex plane $\mathbb{C}$ as Möbius transformations. For $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z})$ and $z \in \mathbb{C}$ the action is

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} .z = \frac{az+b}{cz+d}$$

Möbius transformations are *conformal maps* (i.e. they locally preserve angles). Let $\mathbb{H} = \{\tau \in \mathbb{C} \mid \mathrm{Im}(\tau) > 0\}$ be the upper half-plane of $\mathbb{C}$. One easily checks that for $\tau \in \mathbb{H}$ and $A \in \mathrm{SL}_2(\mathbb{Z})$, $A.\tau$ is also in $\mathbb{H}$ so we have an action of $\mathrm{SL}_2(\mathbb{Z})$ on $\mathbb{H}$. The orbit space $\mathbb{H}/\mathrm{SL}_2(\mathbb{Z})$ parametrizes the *conformal equivalence classes* of tori, as we will see. Tori are conformally equivalent if there is a conformal (complex-analytic) bijection mapping one to the other.

One way to define a torus is as the orbit space of $\mathbb{C}$ by a *lattice*. Let $\tau, w \in \mathbb{C}$ be linearly independent over $\mathbb{R}$ (i.e not real multiples of each other). The lattice with basis τ, w is the $\mathbb{Z}$-span of τ and w, denoted $\mathbb{Z}\tau + \mathbb{Z}w = \{l\tau + mw \mid l, m \in \mathbb{Z}\}$. One can show that the lattice with basis τ, w equals the lattice with basis $a\tau + bw, c\tau + dw$ if and only if $a, b, c, d \in \mathbb{Z}$ with $ad - bc = \pm 1$. That is, the possible ways to change basis are given by $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in M_2(\mathbb{Z})$ with determinant ± 1. The determinant -1 transformations can be written as a determinant 1 transformation followed by interchanging the basis vectors,

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} b & a \\ d & c \end{pmatrix}$$

The ordering of the basis is irrelevant, so we may restrict to determinant 1 i.e. $\mathrm{SL}_2(\mathbb{Z})$. A lattice acts on $\mathbb{C}$ by translation:

$$z.(l\tau + mw) = z + l\tau + mw$$

The orbit space $\mathbb{C}/(\mathbb{Z}\tau + \mathbb{Z}w)$ looks like a parallelogram with vertices at $0,\tau,w$, $\tau + w$. Every point along the side of the parallelogram is in the same orbit as one on the opposite side, so opposite sides are identified ('glued together') and the result is a torus.

Rotation and scaling are conformal maps, so we can rotate and scale the basis vectors τ, w and end up with a conformally equivalent torus. Consequently we may assume $w = 1$ and $\tau \in \mathbb{H}$. So every $\tau \in \mathbb{H}$ corresponds to a torus. But we can change basis with $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z})$ and still have the same lattice: $(\tau, 1) \longmapsto (a\tau + b, c\tau + d)$. Scaling so that the second basis vector in 1, we get that τ and $\frac{a\tau+b}{c\tau+d} = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right).\tau$ describe the same torus. Consequently $\mathbb{H}/\mathrm{SL}_2(\mathbb{Z})$ parametrizes, without redundancy, the conformal equivalence classes of tori and is called the *moduli space* of the torus.

Chapter 2

Introduction to finite group modular data

2.1 Definition of modular data for a finite group

In this section we provide the definition of modular data for a finite group. We define modular data as a representation of $\mathrm{SL}_2(\mathbb{Z})$ on the space $C^0(G_{comm})$. This space was defined in [KSSB99]. There are two bases of interest for $C^0(G_{comm})$, $\mathcal{C}$ and $\mathcal{P}$, giving two matrix representations of modular data. In §2.3 we review the contexts in which finite group modular data arises. In most of the literature modular data appears only as a matrix representation in the $\mathcal{C}$ basis. However, we present modular data first in the $\mathcal{P}$ basis and explicitly derive modular data in the $\mathcal{C}$ basis. This derivation does not seem to be given elsewhere.

2.1.1 The space $C^0(G_{comm})$ and two bases

Let G act on $G \times G$ by simultaneous conjugation, i.e. for $x \in G$ and $(g,h) \in G \times G$

$$x.(g,h) = (xgx^{-1}, xhx^{-1})$$

Define a subset G_{comm} of $G \times G$ as the set of all commuting pairs, $G_{comm} = \{(g,h) \in G \times G \mid gh = hg\}$. Since g and h commute if and only if xgx^{-1} and xhx^{-1} commute, the G-action on $G \times G$ induces a G-action on G_{comm}. We consider the G-orbits of G_{comm} and will write $(g,h) \sim (g',h')$ to denote elements that are in the same orbit.

Fix a set $R = R(G)$ of representatives of the conjugacy classes of G. For each $a \in R$ fix a set R_a of representatives of the conjugacy classes of $C(a)$. Let $\mathcal{R} = \{(a, b_a) \mid a \in R,\ b_a \in R_a\}$.

Lemma 2.1 *$\mathcal{R}$ forms a set of representatives of the G-orbits of G_{comm}. In particular, every $(g,h) \in G_{comm}$ has a representative $(\hat{g}, \hat{h}_g) \in \mathcal{R}$.*

Proof. First, it is clear that no two elements of $\mathcal{R}$ are in the same orbit. Now let $(g,h) \in G_{comm}$. Then there exists $x \in G$ such that $xgx^{-1} = \hat{g} \in R$, so $(g,h) \sim (\hat{g}, xhx^{-1})$. Since $xhx^{-1} \in C(\hat{g})$ there exists $y \in C(\hat{g})$ such that $y(xhx^{-1})y^{-1} = \hat{h}_g \in R_{\hat{g}}$. So we have $(g,h) \sim (\hat{g}, xhx^{-1}) \sim (y\hat{g}y^{-1}, yxhx^{-1}y^{-1}) = (\hat{g}, \hat{h}_g) \in \mathcal{R}$, proving the lemma.

□

Define $C(G_{comm})$ as the space of complex-valued functions on G_{comm} and the subspace $C^0(G_{comm}) \subset C(G_{comm})$ as those functions that are constant on the G-orbits, i.e.

$$C^0(G_{comm}) = \{f \in C(G_{comm}) \mid f(xgx^{-1}, xhx^{-1}) = f(g,h) \text{ for all } x,g,h \in G\}$$

Equivalently, $C^0(G_{comm})$ is the space of complex-valued functions on the orbit space G_{comm}/G.

First we build the basis $\mathcal{P}$ of $C^0(G_{comm})$, called the *permutation basis*. We will see later that modular data is a permutation representation when expressed in this basis, hence the name. For each $a \in R$ and $b_a \in R_a$ define the function $[\![a, b_a]\!]$ to be the characteristic function of the G-orbit of (a, b_a), i.e.

$$[\![a,b_a]\!](g,h) = \begin{cases} 1 & (g,h) \sim (a,b_a) \\ 0 & \text{otherwise} \end{cases}$$

We apologize for the awkward choice of notation $[\![a, b_a]\!]$ for a function. After this section we identify $[\![a, b_a]\!]$ with the G-orbit of (a, b_a). In particular, keep in mind that $[\![a,b]\!] = [\![c,d]\!]$ if and only if there exists $g \in G$ such that $(gag^{-1}, gbg^{-1}) = (c,d)$. For Abelian groups conjugation is trivial so $[\![a,b]\!] = [\![c,d]\!] \iff (a,b) = (c,d)$.

Proposition 2.2 (permutation basis) *The set of functions* $\mathcal{P} = \{[\![a,b_a]\!] \mid a \in R,\ b_a \in R_a\}$ *forms a basis for* $C^0(G_{comm})$.

Proof. Considering $C^0(G_{comm})$ as $\mathbb{C}$-functions on the orbit space G_{comm}/G, the $[\![a, b_a]\!]$ are the characteristic functions of the orbits hence form a basis.

□

Corollary 2.3 *The dimension of* $C^0(G_{comm})$ *is* $\sum_{a \in R} q_a$, *where* q_a *is the number of conjugacy classes of* $C_G(a)$.

We now build the second basis $\mathcal{C}$, called the *character basis* (the name comes from characters of a vertex operator algebra in conformal field theory). For $(g,h) \in G_{comm}$, let $(\hat{g}, \hat{h}_g)$ be its representative in $\mathcal{R}$. For each $a \in R$ and irreducible character $\chi \in \mathrm{Irr}(C_G(a))$, define the function $\mathrm{ch}_a^\chi \in C^0(G_{comm})$ by

$$\mathrm{ch}_a^\chi(g,h) = \begin{cases} \overline{\chi(\hat{h}_g)} & \hat{g} = a \\ 0 & \text{otherwise} \end{cases}$$

The definition depends only on the representative $(\hat{g}, \hat{h}_g)$ hence ch_a^χ is in $C^0(G_{comm})$. The functions do not depend on the choice of representatives R. Indeed, replacing a with xax^{-1} we see that $C(xax^{-1}) = xC(a)x^{-1}$. Then $\chi \in \mathrm{Irr}(C(a))$ corresponds with $\chi^x \in \mathrm{Irr}(C(xax^{-1})) = \mathrm{Irr}(xC(a)x^{-1})$, where for $xhx^{-1} \in xC(a)x^{-1}$, $\chi^x(xhx^{-1}) = \chi(h)$. For this reason we may always assume that $g \in R$, and write $\chi(h)$ instead of $\chi(\hat{h}_g)$. We will often identify ch_a^χ with the pair (a,χ).

Proposition 2.4 (character basis) *The set of functions $\mathcal{C} = \{\mathrm{ch}_a^\chi \mid a \in R,\ \chi \in \mathrm{Irr}(C(a))\}$ forms a basis of $C^0(G_{comm})$.*

Proof. Since the number of conjugacy classes of $C(a)$ is equal to the number of irreducible characters of $C(a)$ we have that the number of functions in $\mathcal{C}$ is $\sum_{a\in R} q_a$, i.e. the dimension of $C^0(G_{comm})$. Now we show that the ch_a^χ are linearly independent. Suppose $\mathrm{ch}_a^\chi = \sum_{(b,\psi)\in\mathcal{C}} \alpha_{(b,\psi)} \mathrm{ch}_b^\psi$ for some $\alpha_{(b,\psi)} \in \mathbb{C}$. Then for every $h \in C(a)$ we have

$$\begin{aligned} \overline{\chi(h)} &= \mathrm{ch}_a^\chi(a,h) \\ &= \sum_{(b,\psi)\in\mathcal{C}} \alpha_{(b,\psi)} \mathrm{ch}_b^\psi(a,h) \\ &= \sum_{(a,\psi)\in\mathcal{C}} \alpha_{(a,\psi)} \mathrm{ch}_a^\psi(a,h) \\ &= \sum_{\psi\in\mathrm{Irr}(C(a))} \alpha_{(a,\psi)} \overline{\psi(h)} \end{aligned}$$

This implies that $\alpha_{(a,\psi)} = 0$ for $\psi \neq \chi$ and $\alpha_{(a,\chi)} = 1$ since the irreducible characters of $C(a)$ are linearly independent. Now let $d \in R$ with $d \neq a$. Then for every $k \in C(d)$

we have

$$\begin{aligned} 0 &= \mathrm{ch}_a^\chi(d,k) \\ &= \sum_{(b,\psi)\in\mathcal{C}} \alpha_{(b,\psi)}\mathrm{ch}_b^\psi(d,k) \\ &= \sum_{(d,\psi)\in\mathcal{C}} \alpha_{(d,\psi)}\overline{\psi(k)} \\ &= \sum_{\psi\in\mathrm{Irr}(C(d))} \alpha_{(d,\psi)}\overline{\psi(k)} \end{aligned}$$

which implies all the $\alpha_{(d,\psi)} = 0$ since the $\psi \in \mathrm{Irr}(C(d))$ are linearly independent. So we have that all the $\alpha = 0$ except $\alpha_{(a,\chi)} = 1$ so the ch_a^χ are linearly independent and $\mathcal{C}$ is a basis of $C^0(G_{comm})$.

□

Next we present the formulas needed to change from one basis to the other. We have not seen it written down anywhere in the literature.

Proposition 2.5 (Change of basis formula) *Let* $\mathrm{ch}_a^\chi \in \mathcal{C}$ *and* $[\![a, b_a]\!] \in \mathcal{P}$. *Then we can express* ch_a^χ *in terms of* $\mathcal{P}$ *and* $f_{(a,b_a)}$ *in terms of* $\mathcal{C}$ *as follows:*

$$\mathrm{ch}_a^\chi = \sum_{b_a\in R_a} \overline{\chi(b_a)}[\![a, b_a]\!] \tag{2.1}$$

$$[\![a, b_a]\!] = \frac{1}{|C_{C_G(a)}(b_a)|}\sum_{\chi\in\mathrm{Irr}(\mathrm{C}(a))} \chi(b_a)\mathrm{ch}_a^\chi \tag{2.2}$$

Proof. Let $(g,h) \in G_{comm}$. Evaluating the right-hand side of 2.1 at (g,h) gives

$$\sum_{b_a\in R_a} \overline{\chi(b_a)}[\![a, b_a]\!](g,h) = \begin{cases} \overline{\chi(h)} & a \sim g \\ 0 & \text{otherwise} \end{cases}$$

which is exactly the definition of $\mathrm{ch}_a^\chi(g,h)$. Evaluating the right-hand side of 2.2 at (g,h) and using the Second Orthogonality Relation (1.14) gives

$$\begin{aligned} \frac{1}{|C_{C_G(a)}(b_a)|}\sum_{\chi\in\mathrm{Irr}(\mathrm{C}(a))} \chi(b_a)\mathrm{ch}_a^\chi(g,h) &= \frac{\delta_{a\sim g}}{|C_{C_G(a)}(b_a)|}\sum_{\chi\in\mathrm{Irr}(\mathrm{C}(a))} \chi(b_a)\overline{\chi(h)} \\ &= \begin{cases} 1 & (a,b_a) \sim (g,h) \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

which is the definition of $[\![a, b_a]\!]$.

□

2.1.2 $\mathrm{SL}_2(\mathbb{Z})$ representation and the definition of modular data

Let $A = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z})$ and $(g,h) \in G_{comm}$. Define a right action of $\mathrm{SL}_2(\mathbb{Z})$ on G_{comm} by

$$(g,h).\begin{pmatrix} a & b \\ c & d \end{pmatrix} = (g^a h^c, g^b h^d)$$

With a bit of writing one checks that this is a right action, noting that commutativity of g with h is required. The action commutes with the action on G on G_{comm}, namely

$$\begin{aligned}(x.(g,h)).\begin{pmatrix} a & b \\ c & d \end{pmatrix} &= ((xgx^{-1})^a(xhx^{-})^c, (xgx^{-1})^b(xhx^{-1})^d)) \\ &= (xg^ah^cx^{-1}, xg^bh^dx^{-1}) \\ &= x.(g^ah^c, g^bh^d) \\ &= x.\left((g,h).\begin{pmatrix} a & b \\ c & d \end{pmatrix}\right)\end{aligned}$$

Consequently we have that $(g,h) \sim (g',h') \iff (g,h).\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \sim (g',h').\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

This action induces a right action on $C^0(G_{comm})$ via

$$\left(f.\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)\right)(x,y) = f\left((x,y).\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)^{-1}\right)$$

where $f \in C^0(G_{comm})$. As $C^0(G_{comm})$ is a $\mathbb{C}$-vector space, this defines a representation of $\mathrm{SL}_2(\mathbb{Z})$.

Definition 2.6 *The representation of* $\mathrm{SL}_2(\mathbb{Z})$ *on the space* $C^0(G_{comm})$ *is called the* modular data associated with G *and will be denoted* $\rho = \rho_G$.

As $\mathrm{SL}_2(\mathbb{Z})$ is generated by s and t, modular data is determined by $\rho_G(s) = S$ and $\rho_G(t) = T$. We will not use different symbols for S and T with respect to the different bases, but the basis will (hopefully) be clear from context.

Modular data in the permutation basis

For the basis functions $[\![g,h]\!] \in \mathcal{P}$, the $\mathrm{SL}_2(\mathbb{Z})$ action is

$$\begin{aligned}\left([\![g,h]\!].\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)\right)(x,y) &= [\![g,h]\!]\left((x,y).\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)^{-1}\right) \\ &= \begin{cases} 1 & \text{if} (g,h) \sim (x,y).\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)^{-1} \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} 1 & \text{if} (g,h).\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \sim (x,y) \\ 0 & \text{otherwise} \end{cases} \\ &= \left([\![g^ah^c, g^bh^d]\!]\right)(x,y)\end{aligned}$$

Consequently we have a (right) action of $\mathrm{SL}_2(\mathbb{Z})$ on $\mathcal{P}$,

$$[\![g,h]\!].\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) = [\![g^a h^c, g^b h^d]\!]$$

and modular data ρ_G is a permutation representation (hence the name permutation basis for $\mathcal{P}$). Note however that the basis $\mathcal{P}$ is not ordered, so to write ρ as matrix representation we must first fix an order on $\mathcal{P}$. We will discuss this in detail in Chapter 4, but for now assume that we fix an order on $\mathcal{P}$.

For the generators $s = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ and $t = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$, we have

$$[\![g,h]\!].s = [\![h,g^{-1}]\!] \tag{2.3}$$

$$[\![g,h]\!].t = [\![g,gh]\!] \tag{2.4}$$

Consequently, S and T in the $\mathcal{P}$ basis are given by

$$S_{(g',h'),(g,h)} = \begin{cases} 1 & (h,g^{-1}) \sim (g',h') \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

$$T_{(g',h'),(g,h)} = \begin{cases} 1 & (g,gh) \sim (g',h') \\ 0 & \text{otherwise} \end{cases} \tag{2.6}$$

Modular data in the character basis

For the character basis, the action of a general element $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z})$ on the basis functions ch_a^χ is not as easy to describe as in the permutation basis. But it suffices to give the action of the generators s and t.

Proposition 2.7 *For $\mathrm{ch}_a^\chi \in C$, the action of s, $t \in \mathrm{SL}_2(\mathbb{Z})$ is described by*

$$\mathrm{ch}_a^\chi.t = \frac{\chi(a)}{\chi(e)}\mathrm{ch}_a^\chi \tag{2.7}$$

$$\mathrm{ch}_a^\chi.s = \frac{1}{|C(a)|}\sum_{(b,\psi)\in C}\frac{1}{|C(b)|}\sum_{x\in G(a,b)}\overline{\chi(xbx^{-1})\psi(x^{-1}ax)}\mathrm{ch}_b^\psi \tag{2.8}$$

where $G(a,b) = \{x \in G \mid axbx^{-1} = xbx^{-1}a\}$.

Remark 2.8 *The set $G(a,b)$ is precisely the set of $x \in G$ such that $x^{-1}ax \in C(b)$ and $xbx^{-1} \in C(a)$, i.e. the x such that $\chi(xbx^{-1})\psi(x^{-1}ax)$ makes sense. When $G(a,b) = \emptyset$ the sum over $G(a,b)$ is 0.*

Proof. The proof of (2.7) is sketched in [CGR00] and we elaborate it here. The proof of (2.8) is our own, though the result is of course the standard definition of S.

Let $(g,h) \in G_{comm}$. We prove (2.7) first. Evaluating the left-hand side of (2.7) at (g,h) gives

$$(\mathrm{ch}_a^\chi.t)(g,h) = \mathrm{ch}_a^\chi(g,g^{-1}h) = \begin{cases} \chi(h^{-1}g) & g \sim a \\ 0 & \text{otherwise} \end{cases}$$

Evaluating the right-hand side gives

$$\frac{\chi(a)}{\chi(e)}\mathrm{ch}_a^\chi(g,h) = \begin{cases} \frac{\chi(a)\chi(h^{-1})}{\chi(e)} & g \sim a \\ 0 & \text{otherwise} \end{cases}$$

We may assume $g = a$, so we need to show $\frac{\chi(g)\chi(h^{-1})}{\chi(e)} = \chi(h^{-1}g)$. Let φ be the representation of $C(g)$ corresponding to χ. Since $g \in Z(C(g))$, Corollary 1.6 gives $\varphi(g) = \xi I$ for some $\xi \in \mathbb{C}$, so $\chi(g) = \deg(\varphi)\xi = \chi(e)\xi$. Then

$$\varphi(h^{-1}g) = \varphi(h^{-1})\varphi(g) = \xi\varphi(h^{-1})$$

hence

$$\chi(h^{-1}g) = \mathrm{tr}(\xi\varphi(h^{-1})) = \xi\chi(h^{-1}) = \frac{\chi(g)}{\chi(e)}\chi(h^{-1})$$

as required.

For (2.8), the left-hand side is

$$(\mathrm{ch}_a^\chi.s)(g,h) = \mathrm{ch}_a^\chi(h^{-1},g) = \begin{cases} \chi(g^{-1}) & h^{-1} \sim a \\ 0 & \text{otherwise} \end{cases}$$

In evaluating the right-hand side, we use the Second Orthogonality Relation (1.14), Lemma 1.1, and Proposition 1.9:

$$\begin{aligned}
&\frac{1}{|C(a)|}\sum_{(b,\psi)\in C}\frac{1}{|C(b)|}\sum_{x\in G(a,b)}\overline{\chi(xbx^{-1})\psi(x^{-1}ax)}\mathrm{ch}_b^\psi(g,h) \\
= {} & \frac{1}{|C(a)||C(g)|}\sum_{\psi\in\mathrm{Irr}(C(g))}\sum_{x\in G(a,g)}\overline{\chi(xgx^{-1})\psi(x^{-1}ax)}\psi(h^{-1}) \\
= {} & \frac{1}{|C(a)||C(g)|}\sum_{x\in G(a,g)}\overline{\chi(xgx^{-1})}\left(\sum_{\psi\in\mathrm{Irr}(C(g))}\overline{\psi(x^{-1}ax)}\psi(h^{-1})\right) \\
= {} & \frac{1}{|C(a)||C(g)|}\sum_{x\in G(a,g)}\overline{\chi(xgx^{-1})}|C_{C(g)}(h^{-1})|\delta_{x^{-1}ax\sim_{C(g)}h^{-1}} \\
= {} & \frac{1}{|C(a)||K_{C(g)}(h^{-1})|}\sum_{x\in X}\chi(xg^{-1}x^{-1})
\end{aligned}$$

where X is the set of all $x \in G$ satisfying

$$axgx^{-1} = xgx^{-1}a \qquad (2.9)$$

$$x^{-1}ax \sim_{C(g)} h^{-1} \qquad (2.10)$$

If (2.10) is satisfied, then we have $z \in C(g)$ such that $zx^{-1}axz^{-1} = h^{-1}$ hence $a \sim_G h^{-1}$. So for $a \not\sim_G h^{-1}$, the sum is empty and the value is 0 as desired. Otherwise, we we may assume $a = h^{-1}$. Since g and h commute, so do g and $h^{-1} = a$ so $a \in C(g)$. We need to know more about the set X.

Partition G into right cosets of $C(a)$. Let $x \in X$, and take any $z \in C(a)$. We show that $zx \in X$:

$$a(zx)g(zx)^{-1} = z(axgx^{-1})z^{-1} = z(xgx^{-1}a)z^{-1} = (zx)g(zx)^{-1}a$$

and

$$(zx)^{-1}azx = x^{-1}z^{-1}azx = x^{-1}ax \sim_{C(g)} a$$

hence $C(a)x \subset X$. So X must be a union of right cosets of $C(a)$.

Now let $C(a)w \subset X$ and suppose that for every choice of coset representative w, $w \notin C(g)$. Since $w \in X$, by (2.10) there exists $v \in C(g)$ such that $vw^{-1}awv^{-1} = a$, i.e. $vw^{-1} \in C(a)$. But this is exactly the condition for the cosets $C(a)w$ and $C(a)v$ to be equal, which is a contradiction since $v \in C(g)$. Hence every coset that is contained in X can be written as $C(a)x$ with $x \in C(g)$. Conversely, every coset $C(a)x$ with $x \in C(g)$ is in X since clearly $C(g) \subset X$. So every element of X can be written as zx with $z \in C(a)$ and $x \in C(g) = C(g^{-1})$. Hence the character value appearing in the sum is

$$\chi((zx)g^{-1}(zx)^{-1}) = \chi(zg^{-1}z^{-1}) = \chi(g^{-1})$$

To complete the proof we need to see that $|X| = |C(a)||K_{C(g)}(a)|$.

Let $K_{C(g)}(a) = \{k_1, k_2, \dots, k_l\}$ and partition $C(g)$ into l subsets $C(g)_i = \{x \in C(g) \mid xax^{-1} = k_i\}$. Note that $C(g)_i \neq \emptyset$. For $x_i \in C(g)_i$ and $x_j \in C(g)_j$ we get

$$\begin{aligned} i = j \quad &\iff \quad x_i a x_i^{-1} = x_j a x_j^{-1} \\ &\iff \quad x_j^{-1} x_i a x_i^{-1} x_j = a \\ &\iff \quad x_j^{-1} x_i \in C(a) \\ &\iff \quad C(a)x_i = C(a)x_j \end{aligned}$$

hence the cosets $C(a)x$ with $x \in C(g)$ are parameterized by the $C(g)_i$ and there are l of them, showing that $|X| = |C(a)||K_{C(g)}(a)|$ and completing the proof.

□

We will use the notation (a, χ) instead of ch_a^χ to index S and T in the character basis. Proposition 2.7 gives the following formulas for S and T. These matrices are the way finite group modular data is usually defined.

$$S_{(a,\chi),(b,\psi)} = \frac{1}{|C_G(a)||C_G(b)|} \sum_{x \in G(a,b)} \overline{\chi(xbx^{-1})\psi(x^{-1}ax)} \tag{2.11}$$

$$T_{(a,\chi),(b,\psi)} = \delta_{a,b}\delta_{\chi,\psi}\frac{\chi(a)}{\chi(e)} \tag{2.12}$$

Observe that S is symmetric and T is diagonal. An alternate way to write S is

$$S_{(a,\chi),(b,\psi)} = \frac{1}{|G|} \sum_{g \in K_a,\ h \in K_b \cap C_G(g)} \overline{\chi(xhx^{-1})\psi(ygy^{-1})} \tag{2.13}$$

where x, y are any solutions to $g = x^{-1}ax$ and $h = y^{-1}by$.

Since $\mathcal{C}$ is not ordered, there are $|\mathcal{C}|!$ possible ways of choosing an ordering so that we can S and T as matrices. We will see in Chapter 4 that there is a natural way to impose some order on $\mathcal{C}$ (not a total order though).

2.2 Properties of modular data

In this section we give a few properties of finite group modular data that will be useful later. We start with two simple observations.

Lemma 2.9 *For any $(a, \chi), (b, \psi) \in \mathcal{C}$, if the order of a does not divide the exponent of $C(b)$ or the order of b does not divide the exponent of $C(a)$ then $S_{(a,\chi),(b,\psi)} = 0$.*

Proof. We prove the contrapositive. If $G(a, b) \neq \emptyset$ then $g^{-1}ag \in C(b)$ and for some $g \in G$. Hence $|a| = |g^{-1}ag|$ divides the exponent of $C(b)$. Similarly $|b|$ divides the exponent of $C(a)$.

□

Lemma 2.10 *The order of T is the exponent of G.*

Proof. The order of a group element is invariant under conjugation, hence the order of T is the same in both the $\mathcal{C}$ and $\mathcal{P}$ bases. We will use the $\mathcal{P}$ basis. Let m be the order of T. Then t^m acts trivially on $\mathcal{P}$. For every $g \in G$, $[\![g, e]\!]$ is in $\mathcal{P}$, so

$$[\![g, g^m]\!] = [\![g, e]\!].t^m = [\![g, e]\!]$$

But $[\![g, g^m]\!] = [\![g, e]\!] \iff g^m = e$, hence $|g|$ divides m. This holds for all $g \in G$, hence $\mathrm{lcm}\{|g| \mid g \in G\} = \mathrm{Exponent}(G)$ divides m. Conversely, if $l = \mathrm{Exponent}(G)$ then $[\![g, h]\!].t^l = [\![g, g^l h]\!] = [\![g, h]\!]$ so l divides m.

□

Next we give a proposition from [CGR00] that will help us compute certain entries of S.

Proposition 2.11 *Let $(a,\chi), (b,v) \in C$ and suppose that v is the restriction to $C(b)$ of some character v' defined on the group $\langle G(a,b), C(b)\rangle$. Then if $G(a,b) \neq \phi$,*

$$S_{(a,\chi),(b,v)} = \frac{\overline{v'(a)\chi^G_{C(a)}(b)}}{|C(b)|}$$

where $\chi^G_{C(a)}$ is the induced character. In particular, when $b = z \in Z(G)$ we get

$$S_{(a,\chi),(z,v)} = \frac{\overline{v(a)\chi(z)}}{|C(a)|} \tag{2.14}$$

Proof. Since v' is defined on $\langle G(a,b), C(b)\rangle$ and restricts to v on $C(b)$, we have for all $g \in G(a,b)$, $v(g^{-1}ag) = v'(g^{-1}ag) = v'(gg^{-1}agg^{-1}) = v'(a)$ so we get

$$\begin{aligned} S_{(a,\chi),(b,v)} &= \frac{1}{|C(a)||C(b)|} \sum_{g \in G(a,b)} \overline{\chi(gbg^{-1})v(g^{-1}ag)} \\ &= \frac{\overline{v'(a)}}{|C(a)||C(b)|} \sum_{\substack{g \in G \\ gbg^{-1} \in C(a)}} \overline{\chi(gbg^{-1})} \\ &= \frac{\overline{v'(a)\chi^G_{C(a)}(b)}}{|C(b)|} \end{aligned}$$

When $b = z \in Z(G)$, we have $C(z) = G$ so $v = v'$ and the condition $gzg^{-1} \in C(a)$ in the definition of the induced character is true for all g. Then the the induced character is given by

$$\chi^G_{C(a)}(z) = \frac{1}{|C(a)|} \sum_{g \in G} \chi(gzg^{-1}) = \frac{|G|\chi(z)}{|C(a)|}$$

and the result follows.

□

Finally, but most importantly, we show that ρ_G is in fact a representation of a finite quotient of $\mathrm{SL}_2(\mathbf{Z})$. This is important since the representation theory of finite groups is considerably simpler than that of infinite (discrete) groups.

Proposition 2.12 *Modular data ρ_G of G is a representation of* $\mathrm{SL}_2(\mathbb{Z}_m)$*, where m is the exponent of G, and decomposes into irreducibles*

$$\rho_G = \bigoplus_i m_i \rho_i$$

where ρ_i are the irreducible representations of $\mathrm{SL}_2(\mathbb{Z}_m)$*.*

Proof. First we show that $\Gamma(m) < \ker(\rho_G)$. Letting $\left(\begin{smallmatrix} 1+am & bm \\ cm & 1+dm \end{smallmatrix}\right) \in \Gamma(m)$, we have that for every $[\![g,h]\!] \in \mathcal{P}$,

$$[\![g,h]\!].\begin{pmatrix} 1+am & bm \\ cm & 1+dm \end{pmatrix} = [\![g^{1+am}h^{cm}, g^{bm}h^{1+dm}]\!] = [\![g,h]\!]$$

hence the action of $\Gamma(m)$ on the permutation basis is trivial and $\Gamma(m) < \ker(\rho_G)$. Consequently, ρ_G induces a representation of $\mathrm{SL}_2(\mathbb{Z})/\Gamma(m) \cong \mathrm{SL}_2(\mathbb{Z}_m)$,

$$A\Gamma(m) \longmapsto \rho_G(A) \quad \text{for } A \in \mathrm{SL}_2(\mathbb{Z})$$

which we also refer to as ρ_G. As it is a representation of the finite group $\mathrm{SL}_2(\mathbb{Z}_m)$, it decomposes into irreducibles (Theorem 1.4).

□

In the next chapter we examine the decomposition of ρ_G into irreducibles.

2.3 Appearances of modular data

Modular data of finite groups arises in several contexts. The most important is in conformal field theory ([Gan05],[CGR00]), though it originally appeared in group representation theory ([Lus79]). It can also be realized as an action of the 3-string braid group.

2.3.1 Conformal field theory

In string theory the basic objects are 1-dimensional *strings* rather than point particles. Rather than a world-line, a string traces out a 2-dimensional *world-sheet*. Conformal field theory (CFT) is a quantum field theory whose symmetries include conformal transformations. When the space-time of the CFT is the world-sheet of a string, the CFT gives important information about the corresponding string theory.

One of the most important cases is when the space-time of the CFT is the torus. As we saw from 1.4, the moduli space of tori is $\mathbb{H}/\mathrm{SL}_2(\mathbb{Z})$. The essential quantity in this CFT is the *partition function* $z(\tau)$ ($\tau \in \mathbb{H}$), which describes how the CFT's state space transforms under the symmetry algebra of the CFT (which is a vertex operator algebra). The partition function is a sesquilinear combination of *characters* $\mathrm{ch}_A(\tau)$, which are indexed by *primary fields* $A \in \phi$. Algebraically, the primary fields index the irreducible representations of the vertex operator algebra and the c_A are the characters of those representations. We will discuss the partition function further in Chapter 3. The characters transform nicely under the action of $\mathrm{SL}_2(\mathbb{Z})$ on $\mathbb{H}$:

$$\begin{aligned}\mathrm{ch}_A(s.\tau) &= \mathrm{ch}_A(-1/\tau) = \sum_{B\in\phi} S_{AB}\mathrm{ch}_B(\tau)\\ \mathrm{ch}_A(t.\tau) &= \mathrm{ch}_A(\tau+1) = \sum_{B\in\phi} T_{AB}\mathrm{ch}_B(\tau)\end{aligned}$$

The matrices S and T define a representation of $\mathrm{SL}_2(\mathbb{Z})$ via $s \longmapsto S$, $t \longmapsto T$, which is called *modular data.* So where do finite groups come in?

The finite group modular data arises as follows. Consider first the string theory having as its space-time a n-torus $\mathbb{R}^n/L_n$, where L_n is a self-dual lattice. The modular data arising in the corresponding CFT is trivial: $S = 1$ and $T = 1$. Now take G to be a subgroup of the automorphism group of L_n and form the orbit space $(\mathbb{R}^n/L_n)/G$, as a manifold (the 'orbifold' construction). For the string theory on $(\mathbb{R}^n/L_n)/G$, the modular data arising in the CFT is the modular data of G, i.e. given by (2.11) and (2.12) ($\mathcal{C}$ is the set of primary fields). Other types of modular data arise from other space-times. For example, when the string theory has a compact Lie group as its space-time the modular data is associated with an affine Kac-Moody algebra.

2.3.2 Braid group

Finite group modular data can be derived from a action of the 3-string braid group B_3 on $G \times G$. The braid group B_3 has the presentation

$$B_3 = \langle \sigma_1, \sigma_2 \mid \sigma_1\sigma_2\sigma_1 = \sigma_2\sigma_1\sigma_2 \rangle$$

Geometrically, consider 3 'strings' (not in the sense of string theory!) going from 3 'start points' to 3 'end points'. The crossings-over and under of the strings and the start/end points that they connect determines the group element (a 'braid'). Two braids are multiplied by joining the end points of one to the start points of the next.

Pretty pictures of braids can be found in any book on the braid group (e.g. [Bir75] or [Kam02]).

$\mathrm{SL}_2(\mathbb{Z})$ arises as a quotient of B_3. The centre of B_3 is $\langle(\sigma_1\sigma_2\sigma_1)^2\rangle$, and $B_3/\langle(\sigma_1\sigma_2\sigma_1)^4\rangle \cong \mathrm{SL}_2(\mathbb{Z})$. The isomorphism is given explicitly by $\sigma_1 \longmapsto t$ and $(\sigma_1\sigma_2\sigma_1)^{-1} \longmapsto s$ (one checks that the relations $s^4 = 1$ and $s^2 = (st)^3$ hold). Modular data comes from an action of B_3 on $G \times G$. Consider the group algebra $\mathbb{C}[G \times G]$ as a $\mathbb{C}$ vector space, and define the action on basis vectors (g, h) by

$$(g,h).\sigma_1 = (g, gh),\ (g,h).\sigma_2 = (gh^{-1}, h)$$

This is a B_3 representation since $\sigma_1\sigma_2\sigma_1(g,h) = \sigma_2\sigma_1\sigma_2(g,h)$. We can get an $\mathrm{SL}_2(\mathbb{Z})$ representation in two ways. First, let V_1 be the subspace of $\mathbb{C}[G \times G]$ spanned by G_{comm} (i.e. by commuting pairs). One checks that V_1 is mapped to itself under the B_3 action and that $(\sigma_1\sigma_2\sigma_1)^4$ acts trivially on V_1, hence we get a representation of $\mathrm{SL}_2(\mathbb{Z})$. The second way is to let V_2 be the subspace spanned by all vectors of the form $\sum_{x\in G}(xgx^{-1}, xhx^{-1})$. Again one checks that V_2 is stable under the B_3 action and that $(\sigma_1\sigma_2\sigma_1)^4$ acts trivially. Now form the intersection $V_1 \cap V_2$ and notice that it is isomorphic to $C^0(G_{comm})$. Notice that σ_1 and $(\sigma_1\sigma_2\sigma_1)^{-1}$ act on $G \times G$ by

$$\begin{aligned}
(g,h).\sigma_1 &= (g, gh)\\
(g,h).(\sigma_1^{-1}\sigma_2^{-1}\sigma_1^{-1}) &= (gh, h).(\sigma_2^{-1}\sigma_1^{-1})\\
&= (gh, h^{-1}g^{-1}h).\sigma_1^{-1}\\
&= (h, h^{-1}g^{-1}h)
\end{aligned}$$

Restricting the action to $V_1 \cap V_2$, this coincides with the action of s and t on $C^0(G_{comm})$.

The action of B_3 on $G \times G = G^2$ generalizes to an action on the n-string braid group B_n on G^{n-1} (and on $\mathbb{C}[G^n]$). The action is given in Chapter 2 of [Gan06]. It plays a role in CFT on higher-genus surfaces (see Chapter 6).

2.3.3 Other appearances

The definition of the S matrix (in the C basis) originates with G.Lusztig in [Lus79], in a purely group-theoretical context. This paper classifies the 'unipotent' representations of certain finite Chevalley groups. Lusztig uses C (as pairs, not functions on $C^0(G_{comm})$) to parametrize the representations. The representations fall into families $\mathcal{F}$, and to each family is associated a finite group $\Gamma_{\mathcal{F}}$ (one of S_1,S_2,S_3,S_5).

Then $\mathcal{C}_{\Gamma_{\mathcal{F}}}$ parametrizes the representations in $\mathcal{F}$. The matrix S appears (though with a slightly different definition) and is used to define a 'Fourier transform' on functions on $\mathcal{C}_\Gamma$.

Modular data of G also arises in the representation theory of the quantum double of G ([KSSB99], [Mas95]). The quantum double $D(G)$ is a ribbon Hopf algebra formed from G. As a vector space, it is identified with the space of $\mathbb{C}$-valued functions on $G \times G$. The irreducible representations of $D(G)$ are indexed by $\mathcal{C}$ (as pairs (a, χ), not functions), with a and $C_G(a)$ being used in the definition of the representation space. Their characters can be identified with $\mathcal{C}$, as functions on $G \times G$. Then the representation ρ_G arises as the action of $\mathrm{SL}_2(\mathbf{Z})$ on $\mathcal{C}$.

2.4 Examples of modular data

We now work out some examples for modular data. Material in this section is based on the presentation given in [CGR00].

2.4.1 Abelian groups

Modular data for Abelian groups is easy to write down. Every Abelian group G can be expressed as $G \cong \mathbf{Z}_{d_1} \times \mathbf{Z}_{d_2} \times \cdots \times \mathbf{Z}_{d_l}$ where $d_1 | d_2 | \cdots | d_l$. Let $m = (m_1, m_2, \ldots, m_l) \in G$. Each conjugacy class consists of a single element, so $R = G$, and all of the centralizers are equal to G. Consequently, the size of $\mathcal{C}$ is $|G|^2$. All the irreducible representations are 1-dimensional, hence are equal to their characters. They are parameterized by $r = (r_1, r_2, \ldots, r_l) \in \mathbf{Z}_{d_1} \times \mathbf{Z}_{d_2} \times \cdots \times \mathbf{Z}_{d_l}$ and are given by

$$\begin{aligned}
\chi_r(m) &= \exp\left(\frac{2\pi i m_1}{d_1}\right)^{r_1} \exp\left(\frac{2\pi i m_2}{d_2}\right)^{r_2} \cdots \exp\left(\frac{2\pi i m_l}{d_l}\right)^{r_l} \\
&= \exp\left(2\pi i \sum_j \frac{m_j r_j}{d_j}\right)
\end{aligned}$$

Then the formulas for S and T in the character basis are

$$\begin{aligned}
S_{(m,r),(m',r')} &= \frac{1}{|G|^2} \sum_{g \in G} \chi_r(m') \chi_{r'}(m) \\
&= \frac{1}{|G|} \exp\left(2\pi i \sum_j \frac{r_j m'_j + r'_j m_j}{d_j}\right) && (2.15) \\
T_{(m,r),(m,r)} &= \exp\left(2\pi i \sum_j \frac{m_j r_j}{d_j}\right) && (2.16)
\end{aligned}$$

2.4.2 Dihedral groups

The dihedral group of order $2n$ is given by the presentation $D_n = \langle x, y \mid x^2 = y^n = e,\ xy = y^{-1}x\rangle$ with elements listed as $x^i y^j$ where $i \in \{0,1\}$, $0 \le j < n$. Results are different for the cases when n is even and when n is odd. First consider the case n even. D_n has $\frac{n}{2}+3$ conjugacy classes. Below is the character table, along with the centralizers of the conjugacy class representatives ($1 \le i, j \le \frac{n}{2}-1$).

D_n	$K(e)$	$K(x)$	$K(xy)$	$K(y^{\frac{n}{2}})$	$K(y^j)$
1	1	1	1	1	1
ψ_1	1	-1	-1	1	1
ψ_2	1	1	-1	$(-1)^{\frac{n}{2}}$	$(-1)^j$
ψ_3	1	-1	1	$(-1)^{\frac{n}{2}}$	$(-1)^j$
χ_i	2	0	0	$2(-1)^i$	$2\cos\left(\frac{2\pi ij}{n}\right)$
$C(g)$	D_n	$\langle x, y^{\frac{n}{2}}\rangle \cong \mathbb{Z}_2 \times \mathbb{Z}_2$	$\langle xy, y^{\frac{n}{2}}\rangle \cong \mathbb{Z}_2 \times \mathbb{Z}_2$	D_n	$\langle y\rangle \cong \mathbb{Z}_n$

From the above table, we can compute the size of $\mathcal{C}$:

$$|\mathcal{C}| = \left(\frac{n}{2}+3\right) + 4 + 4 + \left(\frac{n}{2}+3\right) + \left(\frac{n}{2}-1\right)n = \frac{n^2}{2} + 14$$

The characters of $\langle y\rangle \cong \mathbb{Z}_n$ are $v_i(y^j) = \xi_n^{ij}$, with ξ_n a primitive nth root of unity and $0 \le i < n$. For $\langle x, y^{\frac{n}{2}}\rangle \cong \mathbb{Z}_2 \times \mathbb{Z}_2$ and $\langle xy, y^{\frac{n}{2}}\rangle \cong \mathbb{Z}_2 \times \mathbb{Z}_2$, the characters are $\phi_{rl}(x^j y^{k\frac{n}{2}}) = (-1)^{rj+lk}$ and $\varphi_{rl}((xy)^j y^{k\frac{n}{2}}) = (-1)^{rj+lk}$ respectively.

Since e and $y^{\frac{n}{2}}$ are in $Z(G)$, we can use (2.14) to compute any entry with an index involving e or $y^{\frac{n}{2}}$. Since $\mathbb{Z}_2 \times \mathbb{Z}_2$ has exponent 2 but no element y^i, $1 \le i < \frac{n}{2}$ has order 2, Proposition 2.9 tells us that any entry indexed by y^i and either x or xy is 0. For $S_{(y^k,\chi_i),(y^j,\chi_l)}$, we observe that every conjugate of y^k is either y^k or y^{-k} hence $G(y^k, y^l) = D_n$ and $\chi_i(gy^lg^{-1}) = \chi_i(y^l)$ for every $g \in G$ (similarly for χ_j). So we get

$$S_{(y^k,\chi_i),(y^l,\chi_j)} = \frac{8}{n}\cos\left(\frac{2\pi il}{n}\right)\cos\left(\frac{2\pi jk}{n}\right)$$

Next consider $S_{(x,\phi_{rl}),(x,\phi_{r'l'})}$. Any $x^iy^j \in G(x,x)$ satisfies $xx^iy^jxy^{-j}x^i = x^iy^jxy^{-j}x^ix$, which has solutions $\{y^j, xy^j \mid j = 0, \frac{n}{4}, \frac{n}{2}, \frac{3n}{4}\}$ for $\frac{n}{2}$ even and $\{y^j, xy^j \mid j = 0, \frac{n}{2}\}$ for $\frac{n}{2}$ odd. The character values are now easy to compute, and a similar computation shows that we get the same result for $S_{(xy,\varphi_{rl}),(xy,\varphi_{r'l'})}$, namely

$$S_{(x,\phi_{rl}),(x,\phi_{r'l'})} = S_{(xy,\varphi_{rl}),(xy,\varphi_{r'l'})} = \frac{1}{4}\begin{cases} (-1)^{r+r'+l+l'} + (-1)^{r+r'} & \text{for } \frac{n}{2} \text{ even} \\ (-1)^{r+r'} & \text{for } \frac{n}{2} \text{ odd} \end{cases}$$

Finally, for $S_{(x,\phi_{rl}),(xy,\varphi_{r'l'})}$, we get $G(x,xy) = \{y^j, xy^j \mid j = \frac{n+2}{4}, \frac{3n+2}{4}\}$ when $\frac{n}{2}$ is odd and $G(x,xy) = \emptyset$ when $\frac{n}{2}$ is even. This gives

$$S_{(x,\phi_{rl}),(xy,\varphi_{r'l'})} = \begin{cases} \frac{1}{4}(-1)^{r+r'+l+l'} & \text{for } \frac{n}{2} \text{ odd} \\ 0 & \text{for } \frac{n}{2} \text{ even} \end{cases}$$

For n odd, D_n has $\frac{n-2}{2} + 2$ conjugacy classes. The character table and the centralizers are listed below, where $1 \le i, j \le \frac{n-1}{2}$.

D_n	$K(e)$	$K(x)$	$K(y^j)$
1	1	1	1
ψ_1	1	-1	1
χ_i	2	0	$2\cos\left(\frac{2\pi ij}{n}\right)$
$C(g)$	D_n	$\langle x\rangle \cong \mathbb{Z}_2$	$\langle y\rangle \cong \mathbb{Z}_n$

The size of $\mathcal{C}$ is

$$|\mathcal{C}| = \left(\frac{n-1}{2} + 2\right) + 2 + n\left(\frac{n-1}{2}\right) = \frac{n^2-1}{2} + 4$$

The characters of $\langle y\rangle \cong \mathbb{Z}_n$ are denoted as above, and the characters of $\langle x\rangle \cong \mathbb{Z}_2$ are $\phi_r(x^i) = (-1)^{ri}$, $r \in \{0,1\}$. As before, any entry involving e is computed using (2.14). All the $S_{(x,\phi_r),(y^i,\chi_j)}$ entries are 0 since no y^i has order 2. The $S_{(y^k,\chi_i),(y^l,\chi_j)}$ entries are the same as before. Finally, we have $G(x,x) = \{e, x\}$ hence

$$S_{(x,\phi_r),(x,\phi_{r'})} = \frac{1}{2}(-1)^{r+r'}$$

2.4.3 Quaternion groups

The quaternion group Q_{2n} of order $4n$ is given by the presentation $Q_{2n} = \langle x, y \mid y^{2n} = e,\ x^2 = y^n,\ xy = y^{-1}x\rangle$. Note the similarity with the dihedral groups and recall that for n even, Q_{2n} and D_{2n} have the same character tables. Q_{2n} has $n+3$ conjugacy classes, described earlier in 1.3.5. The centralizers are listed below along with the character table ($\iota = 1$ for n even, $\iota = i$ for n odd).

Q_{2n}	$K(e)$	$K(x)$	$K(xy)$	$K(y^n)$	$K(y^j)$
1	1	1	1	1	1
ψ_1	1	-1	-1	1	1
ψ_2	1	ι	$-\iota$	$(-1)^n$	$(-1)^j$
ψ_3	1	$-\iota$	ι	$(-1)^n$	$(-1)^j$
χ_i	2	0	0	$2(-1)^i$	$2\cos\left(\frac{\pi ij}{n}\right)$
$C(g)$	Q_{2n}	$\langle x\rangle \cong \mathbb{Z}_4$	$\langle xy\rangle \cong \mathbb{Z}_4$	Q_{2n}	$\langle y\rangle \cong \mathbb{Z}_{2n}$

From the above we can compute the size of $\mathcal{C}$:

$$|\mathcal{C}| = (n+3) + 4 + 4 + (n+3) + (n-1)(2n) = 2n^2 + 14$$

The characters of $\langle y\rangle \cong \mathbb{Z}_{2n}$, $\langle x\rangle \cong \mathbb{Z}_4$, and $\langle xy\rangle \cong \mathbb{Z}_4$ are, respectively, $v_i(y^j) = \xi_{2n}^{ij}$, $\phi_r(x^j) = i^{rj}$, and $\varphi_r((xy)^j) = i^{rj}$ where $r \in \{0,1,2,3\}$.

Computing T is straightforward. Computing the entries of S proceeds in a similar manner to the dihedral groups. We use (2.14) for any entries indexed with e or y^n since these are in $Z(G)$. The entries indexed by y^j $(j \neq n)$ and either x or xy are all 0. Indeed, when n is odd, none of the y^j have order 2 or 4, so do not have order dividing $4 = \text{Exponent}(\mathbb{Z}_4)$. When n is even, $y^{\frac{n}{2}}$ and $y^{\frac{3n}{2}}$ have order 4. However, $x^iy^j \in G(x, y^{\frac{n}{2}})$ must satisfy $xx^iy^jy^{\frac{n}{2}}y^{-j}x^{-i} = x^iy^jy^{\frac{n}{2}}y^{-j}x^{-i}x$, which implies $xy^{\frac{n}{2}} = xy^{\frac{3n}{2}}$, a contradiction. Similarly for $y^{\frac{3n}{2}}$.

The entries indexed with y^k and y^l are the same as in the dihedral case, namely

$$S_{(y^k,\chi_i),(y^l,\chi_j)} = \frac{4}{n}\cos\left(\frac{\pi i l}{n}\right)\cos\left(\frac{\pi j k}{n}\right)$$

For $S_{(x,\phi_r),(x,\phi_{r'})}$ we get that $G(x,x) = \{y^j, xy^j \mid j = 0, \frac{n}{2}, n, \frac{3n}{2}\}$ for n even and $G(x,x) = \{y^j, xy^j \mid j = 0, n\}$ for n odd. Then character values are easy to work out, and we get the same result for $S_{(xy,\varphi_r),(xy,\varphi_{r'})}$, namely

$$S_{(x,\phi_r),(x,\phi_{r'})} = S_{(xy,\varphi_r),(xy,\varphi_{r'})} = \begin{cases} \frac{1}{2}\cos\left(\frac{\pi}{2}(r+r')\right) & \text{for } n \text{ even} \\ \frac{1}{4}i^{-r-r'} & \text{for } n \text{ odd} \end{cases}$$

(noting that $2\cos\left(\frac{\pi}{2}(r+r')\right) = (i^{r+r'} + (-i)^{r+r'})$). Finally, for $S_{(x,\phi_r),(xy,\varphi_{r'})}$, we get $G(x,xy) = \{y^j, xy^j \mid j = \frac{n+1}{2}, \frac{3n+1}{2}\}$ when n is odd and $G(x,xy) = \emptyset$ when n is even. This gives

$$S_{(x,\phi_r),(xy,\varphi_{r'})} = \begin{cases} \frac{1}{4}i^{-r-r'} & \text{for } n \text{ odd} \\ 0 & \text{for } n \text{ even} \end{cases}$$

2.4.4 Comparison of dihedral and quaternion modular data

In Chapter 4 we will be comparing modular data of different groups, and defining what we mean by groups having 'equivalent' modular data. To motivate that section, we make a brief comparison of the dihedral and quaternion modular data.

D_{2n} and Q_{2n} have the same character tables when n is even. Their modular data, however, is different. In particular, their T matrices are different. Each $a \in R_{D_{2n}}$ corresponds to an $a' \in R_{Q_{2n}}$ via $x_{D_{2n}} \mapsto x_{Q_{2n}}$, $y_{D_{2n}} \mapsto y_{Q_{2n}}$. Then for $a \neq x, xy$ we easily check that $\chi(a) = \chi'(a')$, hence $T^{D_{2n}}_{(a,\chi),(a,\chi)} = T^{Q_{2n}}_{(a',\chi'),(a',\chi')}$. But in D_{2n} the centralizer of x is $\mathbb{Z}_2 \times \mathbb{Z}_2$ and the four entries $T_{(x,\phi_{rl}),(x,\phi_{rl})}$, $r,l \in \{0,1\}$ are $1,1,-1,-1$, while in Q_{2n} the centralizer of x is $\mathbb{Z}_4$ and the four entries $T_{(x,\phi_r),(x,\phi_r)}$ are $1, i, -1, -i$. Similarly for xy, hence $T^{D_{2n}}$ and $T^{Q_{2n}}$ are different. In particular, for D_4 and Q_4 we have

$$\begin{aligned} T^{D_4} &= \mathrm{diag}(1,1,1,1,1,1,-1,-1,1,1,-1,-\boldsymbol{i},\boldsymbol{i},1,-1,1,-1,1,1,1,1,-1) \\ T^{Q_4} &= \mathrm{diag}(1,1,1,1,1,1,-1,-\boldsymbol{i},\boldsymbol{i},1,-1,-\boldsymbol{i},\boldsymbol{i},1,-1,-\boldsymbol{i},\boldsymbol{i},1,1,1,1,-1) \end{aligned}$$

Trying to compare S matrices is more difficult. We can see that the same entries appear in $S^{D_{2n}}$ and $S^{Q_{2n}}$. But without fixing an order on $\mathcal{C}$, we cannot test for equality. We write down the S matrices for D_4 and Q_4 as an example (see below).

Notice that although the matrices are very similar, the value -4 appears six times in the diagonal of S^{Q_4} but only twice in the diagonal of S^{D_4}. Reordering the basis means simultaneously permuting rows and columns of S, which does not change the entries that appear on the diagonal (they are permuted though). Consequently, we should regard these S matrices as inequivalent. We will define formally what we mean by equivalent in Chapter 4, and have the same entries appearing in the diagonal will be one of the conditions.

$$S^{D_4} = \frac{1}{8}\begin{bmatrix}
1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 2 \\
1 & 1 & 1 & 1 & 2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 2 \\
1 & 1 & 1 & 1 & 2 & -2 & -2 & -2 & -2 & 2 & 2 & 2 & 2 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 2 \\
1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 2 \\
2 & 2 & 2 & 2 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & -2 & -4 \\
2 & -2 & -2 & 2 & 0 & 4 & 0 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -2 & 2 & 0 \\
2 & -2 & -2 & 2 & 0 & 0 & 4 & 0 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 2 & 2 & -2 & 0 \\
2 & -2 & -2 & 2 & 0 & -4 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -2 & 2 & 0 \\
2 & -2 & -2 & 2 & 0 & 0 & -4 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 2 & 2 & -2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 2 & -2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 2 & -2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & -2 & 2 & -2 & 2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 0 & 0 & 0 & 0 & -2 & 2 & -2 & 2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & -4 & 2 & 2 & -2 & -2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 0 & -2 & -2 & 2 & 2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & 0 & -2 & -2 & 2 & 2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 0 & 0 & 4 & 2 & 2 & -2 & -2 & 0 \\
1 & 1 & 1 & 1 & -2 & 2 & -2 & 2 & -2 & 2 & 2 & -2 & -2 & 2 & -2 & -2 & 2 & 1 & 1 & 1 & 1 & -2 \\
1 & 1 & 1 & 1 & -2 & -2 & 2 & -2 & 2 & -2 & -2 & 2 & 2 & 2 & -2 & -2 & 2 & 1 & 1 & 1 & 1 & -2 \\
1 & 1 & 1 & 1 & -2 & -2 & 2 & -2 & 2 & 2 & 2 & -2 & -2 & -2 & 2 & 2 & -2 & 1 & 1 & 1 & 1 & -2 \\
1 & 1 & 1 & 1 & -2 & 2 & -2 & 2 & -2 & -2 & -2 & 2 & 2 & -2 & 2 & 2 & -2 & 1 & 1 & 1 & 1 & -2 \\
2 & 2 & 2 & 2 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & -2 & 4
\end{bmatrix}$$

$$S^{Q_4} = \frac{1}{8}\begin{bmatrix}
1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 2 \\
1 & 1 & 1 & 1 & 2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 2 \\
1 & 1 & 1 & 1 & 2 & -2 & -2 & -2 & -2 & 2 & 2 & 2 & 2 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 2 \\
1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 2 \\
2 & 2 & 2 & 2 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & -2 & -4 \\
2 & -2 & -2 & 2 & 0 & 4 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -2 & 2 & 0 \\
2 & -2 & -2 & 2 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -2 & 2 & 0 \\
2 & -2 & -2 & 2 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 2 & 2 & -2 & 0 \\
2 & -2 & -2 & 2 & 0 & 0 & 0 & 4 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 2 & 2 & -2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 2 & -2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 2 & -2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & -2 & 2 & -2 & 2 & 0 \\
2 & -2 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 0 & 0 & 0 & 0 & -2 & 2 & -2 & 2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 0 & 0 & 2 & 2 & -2 & -2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 2 & 2 & -2 & -2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 4 & -2 & -2 & 2 & 2 & 0 \\
2 & 2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & -2 & -2 & 2 & 2 & 0 \\
1 & 1 & 1 & 1 & -2 & 2 & 2 & -2 & -2 & 2 & 2 & -2 & -2 & 2 & 2 & -2 & -2 & 1 & 1 & 1 & 1 & -2 \\
1 & 1 & 1 & 1 & -2 & -2 & -2 & 2 & 2 & -2 & -2 & 2 & 2 & 2 & 2 & -2 & -2 & 1 & 1 & 1 & 1 & -2 \\
1 & 1 & 1 & 1 & -2 & -2 & -2 & 2 & 2 & 2 & 2 & -2 & -2 & -2 & -2 & 2 & 2 & 1 & 1 & 1 & 1 & -2 \\
1 & 1 & 1 & 1 & -2 & 2 & 2 & -2 & -2 & -2 & -2 & 2 & 2 & -2 & -2 & 2 & 2 & 1 & 1 & 1 & 1 & -2 \\
2 & 2 & 2 & 2 & -4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & -2 & 4
\end{bmatrix}$$

Chapter 3

The centralizer algebra of ρ_G

As we mentioned in §2.3.1, the CFT partition function $z(\tau)$ is a sesquilinear combination of the conformal blocks ch_A. In particular,

$$z(\tau) = \sum_{A,B\in\phi} M_{AB}\mathrm{ch}_A(\tau)\overline{\mathrm{ch}_B(\tau)}$$

We identify z with the matrix M. The values M_{AB} are non-negative integers (they are multiplicities of irreducible representations in the space of states of the CFT). One of the primary fields is distinguished (the 'vacuum'), denoted 0, and $M_{00} = 1$. Since the CFT is symmetric under conformal transformations, the partition function $z(\tau)$ depends only the conformal equivalence class of tori, so is invariant under the $\mathrm{SL}_2(\mathbb{Z})$ action on $\mathbb{H}$, i.e. $z(A.\tau) = z(\tau)$ for $A \in \mathrm{SL}_2(\mathbb{Z})$. From this we derive the most important property of M: M commutes with S and T.

The modular invariants essentially classify the possible CFTs, so one wants to know all the possible modular invariants for given modular data. There will only be finitely many. For modular data arising from affine algebras, some classifications have been done while for finite group modular data, very little is done. Modular invariants have only been computed for a few specific groups (e.g. some dihedrals, A_4, S_3). In theory, one can compute modular invariants for any abelian group but a general formula is not known.

In this section we will not be computing modular invariants, but will be examining the algebra of complex matrices that commute with S and T. We are able to give some general results for the cyclic and dihedral groups, which should help to understand their modular invariants in the general case.

3.1 Permutation representations and the centralizer algebra

The fact that modular data is a permutation representation is tells us much about ρ_G. It seems that this property of modular data has yet to be fully exploited. We start with some background on permutation representations and the centralizer algebra of a representation. Results in this section are standard material and can mostly be found in [Cam99].

Let G act on a finite set Ω. The group action is equivalent to a permutation representation π by taking Ω as a basis for the representation space. The group action induces a component-wise action on $\Omega \times \Omega$,

$$(x, y).g = (x.g, y.g)$$

where $g \in G$ and $(x, y) \in \Omega \times \Omega$. Orbits of this action are called the *orbitals* of G acting on Ω (or orbits of G acting on $\Omega \times \Omega$). The number of orbits and orbitals can be determined from the decomposition of π into irreducibles.

Lemma 3.1 *Let G have permutation representation π corresponding to the action of G on finite set Ω. Let $\pi = \oplus_i m_i \pi_i$ be the decomposition of π into irreducibles with π_1 the trivial representation, and let χ be the character of π. Then*

(a) $\chi(g)$ is the number of fixed points of g.

(b) The number of orbits of G acting on Ω is m_1, the multiplicity of the trivial representation.

(c) The number of orbitals of G acting on Ω is $\sum_i m_i^2$.

(d) The number of orbitals of G acting on Ω is $< \chi, \chi >$.

Proof. Let χ_i be the characters of the π_i. For a permutation representation, the (i, i) matrix entry $\pi(g)_{i,i}$ is 1 if and only if the i^{th} basis element is a fixed point of g. Hence the character value $\chi(g)$, which is the number of 1's on the diagonal of $\pi(g)$, is the number of fixed points of g. Since $\chi(g)$ is an integer, $\chi(g) = \overline{\chi(g)}$. Now

'Burnside's Lemma' gives

$$\begin{aligned}
&\text{number of orbits of } G \text{ acting on } \Omega \\
&= \frac{1}{|G|}\sum_{g\in G}|\mathrm{Fix}(g)| \\
&= \frac{1}{|G|}\sum_{g\in G}1\cdot\chi(g) \\
&= \frac{1}{|G|}\sum_{g\in G}\overline{\chi_1(g)}\chi(g) \\
&= <\chi_1,\chi> \\
&= m_1
\end{aligned}$$

proving the first result. For the third result, first we observe that (ω_1,ω_2) is a fixed point of g in the action on $\Omega\times\Omega$ if and only if both ω_1 and ω_2 are fixed points of g in the action on Ω. Hence $|\mathrm{Fix}_{\Omega\times\Omega}(g)| = |\mathrm{Fix}_{\Omega}(g)|^2$. Then we apply 'Burnside's Lemma' again, giving

$$\begin{aligned}
&\text{number of orbitals of } G \text{ acting on } \Omega \\
&= \text{number of orbits of } G \text{ acting on } \Omega\times\Omega \\
&= \frac{1}{|G|}\sum_{g\in G}|\mathrm{Fix}_{\Omega\times\Omega}(g)| \\
&= \frac{1}{|G|}\sum_{g\in G}|\mathrm{Fix}_{\Omega}(g)|^2 \\
&= \frac{1}{|G|}\sum_{g\in G}\chi(g)^2 \\
&= \frac{1}{|G|}\sum_{g\in G}\overline{\chi(g)}\chi(g) \\
&= <\chi,\chi>
\end{aligned}$$

To get the second result we use the third,

$$\begin{aligned}
<\chi,\chi> &= \frac{1}{|G|}\sum_{g\in G}\overline{\chi(g)}\chi(g) \\
&= \frac{1}{|G|}\sum_{g\in G}\overline{\chi(g)}\left(\sum_i m_i\chi_i(g)\right) \\
&= \sum_i m_i\frac{1}{|G|}\sum_{g\in G}\overline{\chi(g)}\chi_i(g) \\
&= \sum_i m_i<\chi,\chi_i> \\
&= \sum_i m_i^2
\end{aligned}$$

completing the proof.

□

Let $\nu : G \longrightarrow \mathrm{GL}(m, \mathbb{C})$ be a representation of G. Define the *commutant* or *centralizer algebra* $\mathrm{CA}(\nu)$ of ν as the set of $m \times m$ complex matrices that commute with $\nu(g)$ for all $g \in G$, i.e.

$$\mathrm{CA}(\nu) = \{M \in M_m(\mathbb{C}) \mid M\nu(g) = \nu(g)M \quad \text{for all } g \in G\}$$

Notice that $\mathrm{CA}(\nu)$ is closed under addition, matrix multiplication, and scalar multiplication therefore is an algebra over $\mathbb{C}$. Decomposing ν into irreducibles allows us to describe the structure of $\mathrm{CA}(\nu)$, with the next proposition.

Proposition 3.2 *Let ν be a representation of G with decomposition into irreducibles $\nu = \oplus_{i=1}^{l} m_i\nu_i$. Then $\mathrm{CA}(\nu)$ is isomorphic, as a $\mathbb{C}$-algebra, to $\Pi_{i=1}^{l} M_{m_i}(\mathbb{C})$. In particular, the dimension of $\mathrm{CA}(\nu)$ as a $\mathbb{C}$-vector space is $\sum_{i=1}^{l} m_i^2$.*

Proof. Rewrite the decomposition of ν as $\nu = \oplus_{j=1}^{r}\eta_j$, where $r = \sum_{i=1}^{l} m_i$ and the η_j are irreducible (the η_j are the ν_i, just relabeled). We can find a basis such that $\nu(g)$ is a block-diagonal matrix, i.e.

$$\nu(g) = \begin{pmatrix} \eta_1(g) & & & \\ & \eta_2(g) & & \\ & & \ddots & \\ & & & \eta_r(g) \end{pmatrix}$$

The decomposition defines a block-matrix structure on any $r \times r$ matrix, with the (i, j)-block being a $\dim(\eta_i) \times \dim(\eta_j)$ matrix. Let $M \in \mathrm{CA}(\nu)$. Then we have

$$M\nu(g) = \nu(g)M$$

for all $g \in G$. Comparing the (i, j)-blocks of each side of the above gives

$$M_{i,j}\eta_j(g) = \eta_i(g)M_{i,j}$$

for all $g \in G$. For $\eta_i \not\simeq \eta_j$, Schur's Lemma gives $M_{i,j} = 0$ (otherwise $M_{i,j}$ is an isomorphism and η_i and η_j are equivalent). For $\eta_i \simeq \eta_j$, Schur's Lemma gives $M_{i,j} = \lambda_{i,j} I_{\dim(\eta_i)}$ for some $\lambda_{i,j} \in \mathbb{C}$. Notice that for any choice of $\lambda_{i,j}$, $M \in \mathrm{CA}(\nu)$. Hence $\mathrm{CA}(\nu)$ consists of all matrices

$$\begin{pmatrix} A_1 \otimes I_{d_1} & & & \\ & A_2 \otimes I_{d_2} & & \\ & & \ddots & \\ & & & A_l \otimes I_{d_l} \end{pmatrix}$$

where $A_i \in M_{m_i}(\mathbb{C})$ and $d_i = \dim(\nu_i)$. Since $M_{m_i}(\mathbb{C}) \otimes I_{d_i} \cong M_{m_i}(\mathbb{C})$ we have $\mathrm{CA}(\nu) \cong \Pi_i M_{m_i}(\mathbb{C})$. The dimension of $M_{m_i}(\mathbb{C})$ as a vector space is m_i^2 so the result on the dimension of $\mathrm{CA}(\nu)$ follows.

□

Corollary 3.3 *Let π be a permutation representation of G, corresponding to the action of G on Ω. Then the number of orbitals of the action is the dimension of the centralizer algebra* $\mathrm{CA}(\pi)$.

3.2 Decomposing ρ_G

As we saw in the previous section, knowing the decomposition of ρ_G into irreducibles gives the structure of the centralizer algebra of ρ_G, which is a major step in knowing the modular invariants. If we also know the change-of-basis that puts ρ_G into the block-diagonal form then we know the centralizer algebra (the change-of-basis matrix is probably difficult to find however). Just knowing the dimension of the centralizer algebra, however, will constrain the number of modular invariants.

In this section we are able to give the decomposition for the cyclic groups C_p of prime order. We also give the multiplicity of the trivial representation for all cyclic and dihedral groups. The results in this section are original.

Proposition 3.4 *The multiplicity m_1 of the trivial representation in the decomposition of ρ_G is greater than or equal to the number of distinct orders (of elements) in G.*

Proof. The multiplicity m_1 is the number of orbits of $\mathrm{SL}_2(\mathbb{Z})$ acting on $\mathcal{P}$. Let $g, h \in G$ with $|g| \neq |h|$. We claim $[\![g, e]\!]$ and $[\![h, e]\!]$ are in different orbits, proving the proposition. Suppose $[\![g, e]\!]$ and $[\![h, e]\!]$ are in the same orbit. Conjugation does not change the order of an element so $[\![g, e]\!] \neq [\![h, e]\!]$. So there exists $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbf{Z})$ such that

$$[\![g, e]\!] . \begin{pmatrix} a & b \\ c & d \end{pmatrix} = [\![g^a, g^b]\!] = [\![h, e]\!]$$

Then $g^b = e$ so $|g|$ divides b. But a and b are coprime, so a and $|g|$ are coprime, hence $|g| = |g^a| = |h|$ which is a contradiction.

□

Lemma 3.5 *Suppose $g \in G$ has the property that $C(g)$ is cyclic. Then for every $h \in C(g)$ the orbit of $[\![g,h]\!]$ contains an element of the form $[\![a,e]\!]$.*

Proof. Let $C(g) = \langle x \rangle$. Then $g = x^r$ and $h = x^l$ for some $r, l \in \mathbf{Z}$. Let $k = \gcd(r,l)$, and write $k = ar + bl$ so that $1 = a\frac{r}{k} + b\frac{l}{k}$. Then $\left(\begin{smallmatrix} \frac{r}{k} & \frac{l}{k} \\ -b & a \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbf{Z})$ and

$$[\![x^k, e]\!] . \begin{pmatrix} \frac{r}{k} & \frac{l}{k} \\ -b & a \end{pmatrix} = [\![x^r, x^l]\!] = [\![g,h]\!]$$

hence $[\![x^k, e]\!]$ is in the orbit of $[\![g,h]\!]$.

□

Proposition 3.6 *Let ρ_{C_n} be the modular data of C_n. Then then multiplicity m_1 of the trivial representation in the decomposition of ρ_{C_n} is the number of positive divisors of n. Writing $C_n = \langle x \rangle$, the orbit representatives for $\mathrm{SL}_2(\mathbf{Z})$ acting on $\mathcal{P}$ are $[\![x^d, e]\!]$ where d is a positive divisor of n.*

Proof. First we observe that since C_n is Abelian, the $\sim$ relation on G_{comm} is trivial, so $[\![g,h]\!] = [\![g',h']\!] \iff (g,h) = (g',h')$. Lemma 3.5 applies to C_n, hence we need only consider the elements $[\![x^i, e]\!]$, $0 \le i < n$. Clearly $\{[\![e,e]\!]\}$ is one orbit. We claim that for $i, j \neq 0$, $[\![x^i, e]\!]$ and $[\![x^j, e]\!]$ are in the same orbit if and only if $\gcd(i,n) = \gcd(j,n)$. The claim shows that orbit representatives are given by $[\![x^d, e]\!]$ where d is a positive divisor of n, proving the Proposition. Now we prove the claim.

Assume $[\![x^i, e]\!]$ and $[\![x^j, e]\!]$ are in the same orbit. Then there exists $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbf{Z})$ such that

$$[\![x^i, e]\!] . \begin{pmatrix} a & b \\ c & d \end{pmatrix} = [\![x^{ia}, x^{ib}]\!] = [\![x^j, e]\!]$$

i.e. there exist $a, b \in \mathbf{Z}$ coprime and satisfying

$$\begin{aligned} ia &\equiv j \pmod{n} \\ ib &\equiv 0 \pmod{n} \end{aligned}$$

Writing $1 = ka + lb$ and $ib = rn$ for some $k, l, r \in \mathbf{Z}$ gives $i = kai + lrn$. Then for some $\alpha, \beta \in \mathbf{Z}$,

$$\gcd(i,n) = \alpha i + \beta n = (\alpha k)ai + (\alpha lr + \beta)n$$

hence $\gcd(ai,n)$ divides $\gcd(i,n)$. Clearly $\gcd(i,n)$ divides $\gcd(ai,n)$ so $\gcd(ai,n) = \gcd(i,n)$. Now $ai \equiv j \pmod{n}$ implies $\gcd(ai,n) = \gcd(j,n)$, so we get $\gcd(i,n) = \gcd(j,n)$.

Conversely, assume $\gcd(i,n) = \gcd(j,n) = d$. Write $i = dd_i$ and $j = dd_j$. Since d_i and n are coprime, d_i is invertible modulo n, i.e. there exists $d_i^{-1} \in \mathbf{Z}$ and d_i^{-1} is coprime to n. Since d_j is also coprime to n, $\gcd(d_i^{-1}d_j, n) = 1$ hence $\begin{pmatrix} d_i^{-1}d_j & n \\ u & v \end{pmatrix} \in \mathrm{SL}_2(\mathbf{Z})$ for some $u, v \in \mathbf{Z}$. So

$$[\![x^i, e]\!] \cdot \begin{pmatrix} d_i^{-1}d_j & n \\ u & v \end{pmatrix} = [\![x^{dd_id_i^{-1}d_j}, x^{in}]\!] = [\![x^j, e]\!]$$

therefore $[\![x^i, e]\!]$ and $[\![x^j, e]\!]$ are in the same orbit.

□

Proposition 3.7 *Let D_n be the dihedral group of order $2n$ and r the number of positive divisors of n. Then the multiplicity m_1 of the trivial representation in the decomposition of ρ_{D_n} is*

$$m_1 = \begin{cases} r+1, & n \text{ odd} \\ r+3, & n \text{ even}, \frac{n}{2} \text{ odd} \\ r+4, & n \text{ even}, \frac{n}{2} \text{ even} \end{cases}$$

Orbit representatives for $\mathrm{SL}_2(\mathbf{Z})$ acting on $\mathcal{P}$ are as follows, where D is the set of positive divisors of n.

- *For n odd, $\{[\![x,e]\!], [\![y^d, e]\!] \mid d \in D\}$*
- *For n even, $\frac{n}{2}$ even, $\{[\![x,e]\!], [\![xy,e]\!], [\![x, y^{\frac{n}{2}}]\!], [\![xy, y^{\frac{n}{2}}]\!], [\![y^d, e]\!] \mid d \in D\}$*
- *For n even, $\frac{n}{2}$ odd, $\{[\![x,e]\!], [\![xy,e]\!], [\![x, y^{\frac{n}{2}}]\!], [\![y^d, e]\!] \mid d \in D\}$*

Proof. We start with n odd. First, we need to write down the permutation basis, i.e. all the $[\![a, b_a]\!]$ where $a \in R$ and $b_a \in R_a$. The sets R, R_a are summarized below, where i runs through $\{1, 2, \ldots, \frac{n-1}{2}\}$.

$a \in R$	R_a
e	$\{e, x, y^i\}$
x	$\{e, x\}$
y^i	$\{y^j \mid 1 \le j < n\}$

Let $\mathcal{P}'$ be the set of $[\![a, b]\!]$ where x appears in neither a nor b (i.e. when we write a and b in the form x^iy^j with $i \in \{0,1\}$ and $0 \le j < n$, we have $i = 0$). We will show that we can use the result for C_n for these elements. Notice that $\mathcal{P}'$ is a union of $\mathrm{SL}_2(\mathbf{Z})$ orbits (x does not arise as a power of y). Let $\mathcal{P}_{\langle y \rangle} = \{[\![y^i, y^j]\!] \mid 0 \le i, j < n\}$

be the permutation basis of $\langle y\rangle \cong C_n$. If we define the equivalence relation $\bowtie$ on $\mathcal{P}_{\langle y\rangle}$ by

$$[\![y^i, y^j]\!] \bowtie [\![y^k, y^l]\!] \iff (k = n - i \text{ and } l = n - j) \text{ or } (k = i \text{ and } l = j)$$

then we get $\mathcal{P}_{\langle y\rangle}/\bowtie = \mathcal{P}'$ (conjugation by any element involving x takes (y^i, y^j) to (y^{n-i}, y^{n-j})). Now observe that

$$[\![y^i, y^j]\!]. \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} = [\![y^{-i}, y^{-j}]\!] = [\![y^{n-i}, y^{n-j}]\!]$$

meaning two elements related by $\bowtie$ are in the same $\mathrm{SL}_2(\mathbb{Z})$ orbit. Hence the number of orbits of $\mathrm{SL}_2(\mathbb{Z})$ acting on $\mathcal{P}_{\langle y\rangle}$ and $\mathcal{P}_{\langle y\rangle}/\bowtie = \mathcal{P}'$ are the same, namely r. The orbit representatives are the $[\![y^d, e]\!]$ where $d \in D$.

The remaining elements of $\mathcal{P}$ are $[\![x, e]\!]$, $[\![e, x]\!]$, and $[\![x, x]\!]$. These form one orbit, since $[\![x, e]\!].t = [\![x, x]\!]$ and $[\![x, e]\!].s = [\![e, x]\!]$. So the number of orbits is $r + 1$.

Now we turn to n even. The table below describes $\mathcal{P}$, where i runs through $\{1, 2, \ldots, \frac{n}{2} - 1\}$.

$a \in R$	R_a
e	$\{e, x, xy, y^i, y^{\frac{n}{2}}\}$
x	$\{e, x, y^{\frac{n}{2}}, xy^{\frac{n}{2}}\}$
xy	$\{e, xy, y^{\frac{n}{v}}, xyy^{\frac{n}{2}}\}$
y^i	$\{y^j \mid 1 \leq j < n\}$
$y^{\frac{n}{2}}$	$\{e, x, xy, y^i, y^{\frac{n}{2}}\}$

The elements $[\![a, b]\!]$ with x appearing in neither a nor b are handled as in the case n odd, and we get r orbits. The elements $[\![x, e]\!]$, $[\![e, x]\!]$, and $[\![x, x]\!]$ form one orbit, as in the n odd case. Similarly $[\![xy, e]\!]$, $[\![e, xy]\!]$, and $[\![xy, xy]\!]$ form one orbit.

The remaining six elements are $[\![x, y^{\frac{n}{2}}]\!]$, $[\![x, xy^{\frac{n}{2}}]\!]$, $[\![xy, y^{\frac{n}{2}}]\!]$, $[\![xy, xyy^{\frac{n}{2}}]\!]$, $[\![y^{\frac{n}{2}}, x]\!]$, and $[\![y^{\frac{n}{2}}, xy]\!]$. We have $[\![x, y^{\frac{n}{2}}]\!].s = [\![x, xy^{\frac{n}{2}}]\!]$ and $[\![x, y^{\frac{n}{2}}]\!].t = [\![y^{\frac{n}{2}}, x]\!]$, so $[\![x, y^{\frac{n}{2}}]\!]$, $[\![x, xy^{\frac{n}{2}}]\!]$, and $[\![y^{\frac{n}{2}}, x]\!]$ lie in one orbit. Similarly $[\![xy, y^{\frac{n}{2}}]\!]$, $[\![xy, xyy^{\frac{n}{2}}]\!]$, and $[\![y^{\frac{n}{2}}, xy]\!]$ are in the same orbit. For $\frac{n}{2}$ even these are distinct orbits while for $\frac{n}{2}$ they are one orbit. Indeed, suppose $\frac{n}{2}$ is odd and let $\frac{n}{2} - 1 = 2k$. Then

$$[\![y^{\frac{n}{2}}, x]\!].t = [\![y^{\frac{n}{2}}, y^{\frac{n}{2}}x]\!] = [\![y^k y^{\frac{n}{2}} y^{-k}, y^k y^{\frac{n}{2}} x y^{-k}]\!] = [\![y^{\frac{n}{2}}, xy^{\frac{n}{2}-2k}]\!] = [\![y^{\frac{n}{2}}, xy]\!]$$

so we have only one orbit, and the total number of orbits is $r + 3$. Now suppose $\frac{n}{2}$ is even, but that we have only one orbit. Then there exists $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z})$ sending $[\![x, y^{\frac{n}{2}}]\!]$ to $[\![xy, y^{\frac{n}{2}}]\!]$,

$$[\![x, y^{\frac{n}{2}}]\!]. \begin{pmatrix} a & b \\ c & d \end{pmatrix} = [\![x^a y^{c\frac{n}{2}}, x^b y^{d\frac{n}{2}}]\!] = [\![xy, y^{\frac{n}{2}}]\!]$$

Consequently there exists $x^iy^j \in D_n$ such that

$$x^iy^jx^ay^{c\frac{n}{2}}y^{-j}x^i \quad = \quad xy. \tag{3.1}$$

But this implies that $x^{2i+a} = x$, so a is odd and (3.1) becomes $x^ay^{c\frac{n}{c}-2j} = xy$ when i is even, and $x^ay^{-c\frac{n}{2}+2j} = xy$ when i is odd. The former case implies $c\frac{n}{2} - 2j \equiv 1 \pmod{n}$, which is impossible since $\frac{n}{2}$, $-2j$, and n are all even. The latter case gives the same contradiction, so we have two distinct orbits and the total number of orbits is $r + 4$.

□

For the cyclic group C_p, we are able to give the complete decomposition of ρ_{C_p} into irreducible representations of $\mathrm{SL}_2(\mathbb{Z}_p)$. Recall from Theorem 1.10 that the decomposition of ρ_{C_p} into irreducible representations corresponds with the decomposition of its character $\chi_{\rho_{C_p}}$ into irreducible characters. We give the decomposition of the character. The irreducible characters of $\mathrm{SL}_2(\mathbb{Z}_p)$ are described in [Dor71] and [FH91], and we give a review here. A explicit description of the corresponding representations is given in [Eho93].

Let $p \geq 5$ be a prime. $\mathrm{SL}_2(\mathbb{Z}_p)$ has $p + 4$ conjugacy classes, given in the table below. In the proof we will need to know if the trace of the class representative is 2 (note that trace is a class function).

Representative	No. elements in class	No. of classes	Trace
$I = \left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right)$	1	1	2
$z = \left(\begin{smallmatrix}-1&0\\0&-1\end{smallmatrix}\right)$	1	1	-2
$t = \left(\begin{smallmatrix}1&1\\0&1\end{smallmatrix}\right)$	$\frac{p^2-1}{2}$	1	2
$t^2 = \left(\begin{smallmatrix}1&2\\0&1\end{smallmatrix}\right)$	$\frac{p^2-1}{2}$	1	2
$zt = \left(\begin{smallmatrix}-1&-1\\0&-1\end{smallmatrix}\right)$	$\frac{p^2-1}{2}$	1	-2
$zt^2 = \left(\begin{smallmatrix}-1&-2\\0&-1\end{smallmatrix}\right)$	$\frac{p^2-1}{2}$	1	-2
$\left(\begin{smallmatrix}i&0\\0&i^{-1}\end{smallmatrix}\right)$, $i \neq \pm 1$	$p(p+1)$	$\frac{p-3}{2}$	$\neq 2$
$\left(\begin{smallmatrix}i&j\\2j&i\end{smallmatrix}\right)$, $i \neq \pm 1$	$p(p-1)$	$\frac{p-1}{2}$	$\neq 2$

Fix $x \neq 0, \pm 1$ and let $a = \left(\begin{smallmatrix}x&0\\0&x^{-1}\end{smallmatrix}\right)$. Then the elements a^l, for $1 \leq l \leq (p-3)/2$, form a complete set of representatives for the classes of the form $\left(\begin{smallmatrix}i&0\\0&i^{-1}\end{smallmatrix}\right)$. For classes of the form $\left(\begin{smallmatrix}i&j\\2j&i\end{smallmatrix}\right)$, [FH91] shows that a complete set of representatives is given by b^m, where b is an element of order $p+1$ and $1 \leq m \leq (p-1)/2$. Using this notation the character table for $\mathrm{SL}_2(\mathbb{Z}_p)$ is as follows, where ζ_{p-1} and ζ_{p+1} are primitive $p-1$ and $p+1$ roots of unity, $\epsilon = (-1)^{(p-1)/2}$, $1 \le i, l \le (p-3)/2$, and $1 \le j, m \le (p-1)/2$.

$\mathrm{SL}_2(\mathbb{Z}_p)$	$[1]$	$[z]$	$[t]$	$[t^2]$
1	1	1	1	1
ψ	p	p	0	0
χ_i	$p+1$	$(-1)^i(p+1)$	1	1
θ_j	$p-1$	$(-1)^j(p-1)$	-1	-1
ξ_1	$\frac{1}{2}(p+1)$	$\frac{1}{2}\epsilon(p+1)$	$\frac{1}{2}(1+\sqrt{\epsilon p})$	$\frac{1}{2}(1-\sqrt{\epsilon p})$
ξ_2	$\frac{1}{2}(p+1)$	$\frac{1}{2}\epsilon(p+1)$	$\frac{1}{2}(1-\sqrt{\epsilon p})$	$\frac{1}{2}(1+\sqrt{\epsilon p})$
η_1	$\frac{1}{2}(p-1)$	$-\frac{1}{2}\epsilon(p-1)$	$\frac{1}{2}(-1+\sqrt{\epsilon p})$	$\frac{1}{2}(-1-\sqrt{\epsilon p})$
η_2	$\frac{1}{2}(p-1)$	$-\frac{1}{2}\epsilon(p-1)$	$\frac{1}{2}(-1-\sqrt{\epsilon p})$	$\frac{1}{2}(-1+\sqrt{\epsilon p})$

$\mathrm{SL}_2(\mathbb{Z}_p)$	$[zt]$	$[zt^2]$	$[a^l]$	$[b^m]$
1	1	1	1	1
ψ	0	0	1	-1
χ_i	$(-1)^i$	$(-1)^i$	$\zeta_{p-1}^{il}+\zeta_{p-1}^{-il}$	0
θ_j	$(-1)^{j+1}$	$(-1)^{j+1}$	0	$-\zeta_{p+1}^{jm}-\zeta_{p+1}^{-jm}$
ξ_1	$\frac{1}{2}\epsilon(1+\sqrt{\epsilon p})$	$\frac{1}{2}\epsilon(1-\sqrt{\epsilon p})$	$(-1)^l$	0
ξ_2	$\frac{1}{2}\epsilon(1-\sqrt{\epsilon p})$	$\frac{1}{2}\epsilon(1+\sqrt{\epsilon p})$	$(-1)^l$	0
η_1	$\frac{1}{2}\epsilon(1-\sqrt{\epsilon p})$	$\frac{1}{2}\epsilon(1+\sqrt{\epsilon p})$	0	$(-1)^{m+1}$
η_2	$\frac{1}{2}\epsilon(1+\sqrt{\epsilon p})$	$\frac{1}{2}\epsilon(1-\sqrt{\epsilon p})$	0	$(-1)^{m+1}$

When $p = 3$ the character table is the same except there are no $[a^l]$ conjugacy classes and no χ_i characters. When $p = 2$, $\mathrm{SL}_2(\mathbb{Z}_2) \cong S_3$ and the character table is well-known.

Theorem 3.8 *Let C_p be the cyclic group of prime order $p \ge 5$ and χ the character of its modular data ρ_{C_p}. Then χ decomposes as*

$$\chi = 2(1) + \xi_1 + \xi_2 + \psi + 2\sum_{i=1}^{\frac{1}{2}(p-3)} \chi_i$$

For $p = 3$ the decomposition is

$$\chi = 2(1) = \xi_1 + \xi_2 + \psi$$

and for $p = 2$

$$\chi = 2(1) = \hat{\psi}$$

where $\hat{\psi}$ is the 2-dimensional character of $\mathrm{SL}_2(\mathbb{Z}_2) \cong S_3$.

Proof. Assume $p \geq 5$. From Theorem 1.10, we know that the multiplicity of $\nu \in \mathrm{Irr}(\mathrm{SL}_2(\mathbf{Z}_p))$ in the decomposition of χ is given by $< \chi, \nu >$. We need to know the character values $\chi\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$, which by Lemma 3.1 is the number of fixed points of $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ acting on $\mathcal{P}$.

Let $C_p = \langle x \rangle$. Since conjugation in C_p is trivial, elements $[\![x^i, x^j]\!] \in \mathcal{P}$ correspond with vectors $(i,j) \in \mathbf{Z}_p \times \mathbf{Z}_p$ and the action of $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbf{Z}_p)$ on $\mathcal{P}$ corresponds with right multiplication of (i,j) by $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$,

$$[\![x^i, x^j]\!] . \begin{pmatrix} a & b \\ c & d \end{pmatrix} = [\![x^{ai+cj}, x^{bi+dj}]\!] \longleftrightarrow (i,j)\begin{pmatrix} a & b \\ c & d \end{pmatrix} = (ai+cj, bi+dj)$$

Consequently, $[\![x^i, x^j]\!]$ is a fixed point if and only if (i,j) is an eigenvector of $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ with eigenvalue 1. So $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ has a non-trivial fixed point if and only if

$$\begin{aligned} 0 &= \det\begin{pmatrix} a-1 & b \\ c & d-1 \end{pmatrix} \\ 0 &= ad - a - d + 1 - cb \\ 2 &= a + d \\ 2 &= \mathrm{tr}\begin{pmatrix} a & b \\ c & d \end{pmatrix} \end{aligned}$$

Since $\mathbf{Z}_p$ is a field of p elements, the number of fixed points of $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ is p^r, where r is the dimension of the eigenspace. Since $\det\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) = 1$, if 1 is an eigenvalue then it is the only eigenvalue. Hence $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ has a 2-dimensional eigenspace iff $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ is diagonalizable iff $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$. So we have the following formula for χ:

$$\chi\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{cases} 1 & a+d \neq 2 \\ p & a+d = 2,\ \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \neq I \\ p^2 & \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) = I \end{cases} \tag{3.2}$$

Now we can compute the multiplicities. We know from Proposition 3.6 that $m_1 = 2$. Let $\nu \in \mathrm{Irr}(\mathrm{SL}_2(\mathbf{Z}_p))$, $\nu \neq 1$. The multiplicity m_ν of ν in χ is $< \nu, \chi >$. Using (3.2), $< 1, \nu >= 0$ (First Orthogonality Relation), $|\mathrm{SL}_2(\mathbf{Z}_p) = p^3 - p|$, and the fact that χ is 1 except on the conjugacy classes $[1]$,$[t]$, and $[t^2]$, we get the following simplification of $<\chi,\nu>$:

$$\begin{aligned}
<\chi,\nu> &= \frac{1}{|\mathrm{SL}_2(\mathbb{Z}_p)|}\sum_{A\in\mathrm{SL}_2(\mathbb{Z}_p)}\chi(A)\overline{\nu(A)} \\
&= \frac{1}{|\mathrm{SL}_2(\mathbb{Z}_p)|}\left(\sum_{A\in\mathrm{SL}_2(\mathbb{Z}_p)}(\chi(A)-1)\,\overline{\nu(A)}\right)+\frac{1}{|\mathrm{SL}_2(\mathbb{Z}_p)|}\sum_{A\in\mathrm{SL}_2(\mathbb{Z}_p)}\overline{\nu(A)} \\
&= \frac{1}{|\mathrm{SL}_2(\mathbb{Z}_p)|}\left(\sum_{A\in\mathrm{SL}_2(\mathbb{Z}_p)}(\chi(A)-1)\overline{\nu(A)}\right)+<1,\nu> \\
&= (p^3-p)^{-1}(p^2-1)\nu(I)+(p-1)\frac{1}{2}(p^2-1)(\overline{\nu(t)}+\overline{\nu(t^2)}) \\
&= (p^3-p)^{-1}\left((p^2-1)\nu(I)+\frac{1}{2}(p^3-p^2-p+1)(\overline{\nu(t)}+\overline{\nu(t^2)})\right)
\end{aligned}$$

Now computing multiplicities is easy:

$$\begin{aligned}
m_\psi &= (p^3-p)^{-1}\left((p^2-1)p+0+0)\right)=1 \\
m_{\chi_i} &= (p^3-p)^{-1}\left((p^2-1)(p+1)+\frac{1}{2}(p^3-p^2-p+1)2\right)=2 \\
m_{\xi_1} &= (p^3-p)^{-1}\left((p^2-1)\frac{1}{2}(p+1)+\frac{1}{2}(p^3-p^2-p+1)(1)\right)=1 \\
m_{\xi_2} &= m_{\xi_1}=1
\end{aligned}$$

We know that χ has dimension $\chi(I)=p^2=|C|$. Using the multiplicities we know so far and evaluating at I we find that

$$2+\psi(I)+\xi_1(I)+\xi_2(I)+2\sum_{i=1}^{(p-3)/2}(I)=2+p+p+1+(p-3)(p+1)=p^2$$

hence all the remaining multiplicities are 0.

The case $p=3$ is the same except there are no χ_i, and $p=2$ is easy to work out since $\mathrm{SL}_2(\mathbb{Z}_2)\cong S_3$.

$\square$

Knowing the decomposition allows us, by Proposition 3.2, to compute the dimension of the centralizer algebra:

$$\dim \mathrm{CA}(\rho_{\mathbb{Z}_n})=\sum_i m_i^2=4+1+1+1+2(p-3)=2p+1$$

We will obtain the general result for $\mathbb{Z}_n$ in the next section, by counting orbitals.

3.2.1 Galois symmetry

There is a Galois symmetry that might be useful in finding the decomposition for other groups. A representation φ of G is always equivalent to a matrix representation with matrix entries in the cyclotomic field $\mathbb{Q}(\xi_m)$, where m is the exponent of G and ξ_m a primitive m^{th} root of unity. The Galois group $\text{Gal}(\mathbb{Q}(\xi_m)/\mathbb{Q})$ consists of the automorphisms σ_l, $l \in \mathbf{Z}_m^\times$ (the group of units of $\mathbf{Z}_m$), defined by $\sigma_l(\xi_m^s) = \xi_m^{sl}$ and extended linearly to all of $\mathbb{Q}(\xi_m)$. Each automorphism yields a representation $\sigma_l\varphi$ defined by

$$(\sigma_l\varphi(g))_{ij} = \sigma_l(\varphi(g)_{ij})$$

The representations φ and $\sigma_l\varphi$ are said to be in the same *Galois orbit*. For example, the irreducible representations of $\mathbf{Z}_p$ are given by $\chi_i(r) = \xi_p^{ir}$ for some p^{th} root of unity ξ_p and $0 \leq i < p$. All except χ_0 are in the Galois orbit of χ_1 since $\sigma_i\chi_1(r) = \sigma_i(\xi_p^r) = \xi_p^{ir}$.

Since the character of a representation is the trace, the character of $\sigma_l\varphi$ is just $\sigma_l(\chi(g))$, where χ is the character of φ. When χ_i is irreducible, so is $\sigma_l\chi_i$ (we have $<\sigma_l\chi_i, \sigma_l\chi_i> = <\chi_i, \chi_i>$). Now in the case of modular data ρ with character χ we know we know that the character values are integers so $\varphi_l\chi = \chi$. From this we determine that irreducible characters in the same Galois orbit have the same multiplicity in the decomposition of χ. Indeed,

$$\begin{aligned} m_{\chi_i} &= \sigma_l(m_{\chi_i}) \\ &= \sigma_l\left(\frac{1}{|G|}\sum_{g\in G}\chi_i(g)\chi(g)\right) \\ &= \frac{1}{|G|}\sum_{g\in G}\sigma_l\chi_i(g)\chi(g) \\ &= <\sigma_l\chi_i, \chi> = m_{\sigma_l\chi_i} \end{aligned}$$

In the decomposition of $\rho_{\mathbf{Z}_p}$, one finds that the χ_i are in the same Galois orbit as are ξ_1 and ξ_2, and indeed the multiplicities are the same. In this case the multiplicities were easy to work out, but in more difficult cases the Galois symmetry might prove useful.

3.3 Dimension of CA$(\rho_{\mathbf{Z}_n})$

We find the dimension of the centralizer algebra of $\mathbf{Z}_n$, and list the dimensions for some small-order dihedral groups. The results in this section are new. We start with the dimension for $\mathbb{Z}_p$, p a prime, as a stepping stone to the proof for $\mathbb{Z}_n$.

Lemma 3.9 *The number of orbitals for* $\mathrm{SL}_2(\mathbb{Z})$ *acting on the permutation basis* $\mathcal{P}$ *of* $\mathbb{Z}_p$ *is* $2p+1$. *Orbitals representatives are given by*

(a) $([\![1,0]\!],[\![i,0]\!]),\ \ 0 \le i < p$

(b) $([\![1,0]\!],[\![0,j]\!]),\ \ 1 \le j < p$

(c) $([\![0,0]\!],[\![1,0]\!])$

(d) $([\![0,0]\!],[\![0,0]\!])$

Proof. By Proposition 3.6, there are only 2 orbits of $\mathrm{SL}_2(\mathbb{Z}_p)$ acting on $\mathcal{P}$, with representatives $[\![0,0]\!]$ and $[\![1,0]\!]$. Hence every orbital has a representative with first component $[\![0,0]\!]$ or $[\![1,0]\!]$. There are two orbitals with first component $[\![0,0]\!]$, namely $\{([\![0,0]\!],[\![0,0]\!])\}$ and $\{([\![0,0]\!],[\![g,h]\!]) \mid [\![g,h]\!] \in \mathcal{P}\}$. So for the remaining orbitals we may assume there is a representative with first component $[\![1,0]\!]$. The number of orbitals with first component $[\![1,0]\!]$ is equal to the number of orbits of the stabilizer of $[\![1,0]\!]$ acting on $\mathcal{P}$. This stabilizer is given by

$$\begin{aligned}
\mathrm{Stab}_{\mathrm{SL}_2(\mathbb{Z}_p)}([\![1,0]\!]) &= \left\{\left(\begin{smallmatrix} a & b \\ c & d\end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z}_p) \mid [\![a,b]\!] = [\![1,0]\!]\right\} \\
&= \left\{\left(\begin{smallmatrix} a & b \\ c & d\end{smallmatrix}\right) \in \mathrm{SL}_2(\mathbb{Z}_p) \mid a \equiv_p 1,\ b \equiv_p 0\right\} \\
&= \left\{\left(\begin{smallmatrix} 1 & 0 \\ c & 1\end{smallmatrix}\right) \mid c \in \mathbb{Z}_p\right\}
\end{aligned}$$

Elements of $\mathrm{Stab}([\![1,0]\!])$ act on $[\![i,0]\!]$ and $[\![0,i]\!]$ by

$$[\![i,0]\!].\begin{pmatrix} 1 & 0 \\ c & 1\end{pmatrix} = [\![i,0]\!] \tag{3.3}$$

$$[\![0,j]\!].\begin{pmatrix} 1 & 0 \\ c & 1\end{pmatrix} = [\![jc,j]\!] \tag{3.4}$$

From (3.3), we get that no two of $([\![1,0]\!],[\![i,0]\!])$, $0 \le i < p$ are in the same orbital. From (3.4) we see that $A \in \mathrm{Stab}([\![1,0]\!])$ cannot change the second component of $[\![0,i]\!]$, hence $([\![1,0]\!],[\![0,j]\!])$, $1 \le j < p$ are all in different orbitals, which are also different from the orbitals containing the $([\![1,0]\!],[\![i,0]\!])$. Finally, for any $([\![1,0]\!],[\![i,j]\!])$ with $j \neq 0$, j has an inverse $j^{-1} \in \mathbb{Z}_p$ so

$$([\![1,0]\!],[\![i,j]\!]).\begin{pmatrix} 1 & 0 \\ -ij^{-1} & 1\end{pmatrix} = ([\![1,0]\!],[\![i-jij^{-1},j]\!]) = ([\![1,0]\!],[\![0,j]\!])$$

hence $([\![1,0]\!],[\![i,j]\!])$ is in the same orbital as $([\![1,0]\!],[\![0,j]\!])$. Therefore all of the orbitals are accounted for, and there are $2p+1$ of them.

□

For the decomposition of $\rho_{\mathbb{Z}_n}$ we will write $\mathbb{Z}_n$ as a direct product of cyclic groups of prime power order. The modular data of a direct product is just the tensor product:

Lemma 3.10 *Let ρ_G and ρ_H be modular data of G and H. Then the modular data of $G \times H$ is the tensor product $\rho_G \otimes \rho_H$. In particular, in matrix form the modular data is the Kronecker product,*

$$\begin{aligned} S^{G\times H} &= S^G \otimes S^H \\ T^{G\times H} &= T^G \otimes T^H \end{aligned}$$

Proof. This is a straightforward computation involving unpleasant notation, but we will sketch it. Conjugacy classes of $G \times H$ are given by $K_i \times L_j$, where K_i runs through the conjugacy classes of G and L_j through those of H. Representatives are (k_i, l_j) and the centralizers are $C_{G\times H}(k_i, l_j) = C_G(k_i) \times C_H(l_j)$. Irreducible characters of $C_G(k_i) \times C_H(l_j)$ are $\chi_{i,m_i} \cdot \psi_{j,n_j}(g,h) = \chi_{i,m_i}(g)\psi_{j,n_j}(h)$ where $\chi_{i,m_i} \in \mathrm{Irr}(C_G(k_i))$ and $\psi_{j,n_j} \in \mathrm{Irr}(C_H(l_j))$. Then for T we have

$$\begin{aligned} &T^{G\times H}_{((k_i,l_j),\chi_{i,m_i}\psi_{j,n_j}),((k_a,l_b),\chi_{a,m_a}\psi_{b,n_b})} = \\ &\qquad \delta_{(k_i,l_j),(k_a,l_b)}\delta_{\chi_{i,m_i}\psi_{j,n_j},\chi_{a,m_a}\psi_{b,n_b}} \frac{\chi_{i,m_i}(k_i)\psi_{j,n_j}(l_j)}{\chi_{i,m_i}(e)\psi_{j,n_j}(e)} \\ &= \left(\delta_{k_i,k_a}\delta_{\chi_{i,m_i},\chi_{a,m_a}} \frac{\chi_{i,m_i}(k_i)}{\chi_{i,m_i}(e)}\right)\left(\delta_{l_j,l_b}\delta_{\psi_{j,n_j},\psi_{b,n_b}} \frac{\psi_{j,n_j}(l_j)}{\psi_{j,n_j}(e)}\right) \\ &= T^G_{(k_i,\chi_{i,m_i}),(k_a,\chi_{a,m_a})} T^H_{(l_j,\psi_{j,n_j}),(l_b,\psi_{b,n_b})} \end{aligned}$$

hence $T^{G\times H} = T^G \otimes T^H$. The computation for S is similar and more unpleasant.

□

At this point one should ask if the modular data of the semi-direct product $G \ltimes H$ is related to that of G and H. We do not know the answer. For example, D_n is the semi-direct product of C_n and C_2. But the dimensions of their modular data are $\dim(\rho_{C_n}) = n^2$, $\dim(\rho_2) = 4$, $\dim(\rho_{D_n}) = n^2/2 + 14$ (n even), and $\dim(\rho_{D_n}) = (n^2-1)/2+4$ (n odd). Unfortunately, the dimensions do not (at least to this author) suggest any obvious relation.

Proposition 3.11 *Let $n = p_1^{n_1} p_2^{n_2} \cdot \ldots \cdot p_r^{n_r} \in \mathbb{Z}_{>0}$. Then the dimension of the centralizer algebra for $\rho_{\mathbb{Z}_n}$ is*

$$\dim \mathrm{CA}(\rho_{\mathbb{Z}_n}) = \prod_{i=1}^{r} \left((n_i + 1)p_i^{n_i} + n_i p_i^{n_i - 1} \right)$$

Proof. First we show that it suffices to prove the theorem for $\mathbb{Z}_{p^n}$. Let $D(n) = \dim \mathrm{CA}(\rho_{C_n})$. We need to prove that D is multiplicative, i.e. $D(lm) = D(l)D(m)$ when $\gcd(l, m) = 1$. Let $\rho_{\mathbb{Z}_l} = \oplus_i l_i \rho_i$ and $\rho_{\mathbb{Z}_m} = \oplus_j m_j \rho'_j$ be the decompositions of the modular data of $\mathbb{Z}_l$ and $\mathbb{Z}_m$ (note that ρ_i are the irreducible representations of $\mathrm{SL}_2(\mathbb{Z}_l)$ while ρ'_j are the irreducible representations of $\mathrm{SL}_2(\mathbb{Z}_m)$). From Lemma 3.10 we know that the modular data of $\mathbb{Z}_{lm} \cong \mathbb{Z}_l \times \mathbb{Z}_m$ is the tensor product,

$$\begin{aligned}
\rho_{\mathbb{Z}_l \times \mathbb{Z}_m} &= \rho_{\mathbb{Z}_l} \otimes \rho_{\mathbb{Z}_m} \\
&= \left(\bigoplus_i l_i \rho_i \right) \otimes \left(\bigoplus_j m_j \rho'_j \right) \\
&= \bigoplus_{i,j} l_i m_j \rho_i \otimes \rho'_j
\end{aligned}$$

From Theorem 1.13 we know that the irreducible representations of $\mathrm{SL}_2(\mathbb{Z}_{lm}) \cong \mathrm{SL}_2(\mathbb{Z}_l) \times \mathrm{SL}_2(\mathbb{Z}_m)$ are precisely the $\rho_i \otimes \rho'_j$, so the above is the decomposition into irreducibles. Applying Proposition 3.2 we get

$$D(lm) = \sum_{i,j} (l_i m_j)^2 = \left(\sum_i l_i^2 \right) \left(\sum_j m_j^2 \right) = D(l)D(m)$$

as desired.

Now we show that $D(p^n) = (n+1)p^n + np^{n-1}$, by induction on n. The base case $n = 1$ is Lemma 3.9. Assume the formula hold for $n - 1$. Recall the action of $\mathrm{SL}_2(\mathbb{Z})$ on $\mathcal{P}$:

$$[\![x, y]\!] . \begin{pmatrix} a & b \\ c & d \end{pmatrix} = [\![ax + cy, bx + dy]\!]$$

If $x, y \equiv_p 0$ then $ax + cy, bx + dy \equiv_p 0$. Since we have a group action, if at least one of x, y is not 0 modulo p, then so is at least one of $ax + cy, bx + dy$. Consequently we can partition the orbitals into three sets:

- A: the orbitals whose representatives $([\![w, x]\!], [\![y, z]\!])$ have $w, x, y, z \equiv_p 0$.
- B: the orbitals whose representatives $([\![w, x]\!], [\![y, z]\!])$ have at least one of w, x not 0 modulo p

- C: the orbitals whose representatives $([\![w,x]\!],[\![y,z]\!])$ have $w,x \equiv_p 0$ and at least one of y,z not 0 modulo p.

For A, the injective homomorphism $\varphi : \mathbb{Z}_{p^{n-1}} \hookrightarrow \mathbb{Z}_{p^n}$, $x \longrightarrow px$, induces a bijection between A and the orbitals of $\mathrm{SL}_2(\mathbb{Z})$ acting on $\mathcal{P}_{\mathbb{Z}_{p^{n-1}}}$. Then by induction, there are $D(p^{n-1}) = np^{n-1} + (n-1)p^{n-2}$ orbitals in A.

Next we count the orbitals in B. Let $([\![w,x]\!],[\![y,z]\!])$ be any orbital representative from a B orbital. Since either w or x is not 0 modulo p, we know from Proposition 3.6 that $[\![w,x]\!]$ is in the same $\mathrm{SL}_2(\mathbb{Z})$ orbit as $[\![1,0]\!]$. So we may assume the first component is $[\![1,0]\!]$. As in Proposition 3.9, we find that the stabilizer of $[\![1,0]\!]$ is

$$\mathrm{Stab}_{\mathrm{SL}_2(\mathbb{Z}_{p^n})}([\![1,0]\!]) = \left\{ \begin{pmatrix} 1 & 0 \\ c & 1 \end{pmatrix} \mid c \in \mathbb{Z}_{p^n} \right\}$$

Since the second component $[\![y,z]\!]$ is unconstrained, the number of orbitals in B is the number of orbits of the stabilizer on $\mathcal{P}_{\mathbb{Z}_{p^n}}$. The action of the stabilizer is

$$[\![y,z]\!] \cdot \begin{pmatrix} 1 & 0 \\ c & 1 \end{pmatrix} = [\![y+cz,z]\!]$$

In particular, the second component (ie. z) is invariant. Let $z = p^k \hat{z}$, where p^k is the greatest power of p dividing z. Then $\hat{z}^{-1}$ exists modulo p^n. Letting $c = c'\hat{z}^{-1}$ we see that the orbits are given by $\{[\![y + c'p^k, z]\!] \mid c' \in \mathbb{Z},\ z \in \mathbb{Z}_{p^n}\}$, i.e. parametrized by z and the value of y modulo p^k. For $z \equiv_{p^n} 0$ there are p^n orbits, each $[\![y,0]\!]$ in its own orbit for $y \in \mathbb{Z}_{p^n}$. Now assume $z \not\equiv_{p^n} 0$. For $k = 0,1,\ldots,n-1$, there are $p^{n-k} - p^{n-k-1}$ choices for z such that p^k is the greatest power of p dividing z. For such z, each value of y modulo p^k gives a different orbit, so there are $(p^{n-k} + p^{n-k-1})p^k = p^n - p^{n-1}$ orbits for each k. Then the number of orbits in B is

$$p^n + \sum_{k=0}^{n-1} (p^n - p^{n-1}) = (n+1)p^n - np^{n-1}$$

Finally, we count the orbitals in C. Since at least one y,z is not 0 modulo p, we may assume the second component is $[\![1,0]\!]$. Since we assume $w,x \equiv_p 0$, we may use the homomorphism φ to identify $[\![w,x]\!] \in \mathcal{P}_{\mathbb{Z}_{p^n}}$ with $[\![\varphi^{-1}(w),\varphi^{-1}(x)]\!] \in \mathcal{P}_{\mathbb{Z}_{p^{n-1}}}$. So the number of orbitals in C is the number of orbits of $\mathrm{Stab}_{\mathrm{SL}_2(\mathbb{Z})}([\![1,0]\!])$ acting on $\mathcal{P}_{\mathbb{Z}_{p^{n-1}}}$. We know this from the B case above, i.e.

$$np^{n-1} - (n-1)p^{n-2}$$

Then the total number of orbitals is the sum of those in A,B, and C:

$$\begin{aligned} D(p^n) &= (np^{n-1} + (n-1)p^{n-2}) + ((n+1)p^n - np^{n-1}) + (np^{n-1} - (n-1)p^{n-2}) \\ &= (n+1)p^n + np^{n-1} \end{aligned}$$

□

The next step is to find the dimension of the centralizer algebra for the dihedral groups. We have looked briefly at this, but have not found a general formula. Using GAP we computed the dimensions up to D_{20} (via the formula $m_i =< \chi, \chi_i >$). The results are below.

n	$\dim \mathrm{CA}(\rho_{D_n})$	n	$\dim \mathrm{CA}(\rho_{D_n})$
1	5	11	19
2	51	12	186
3	11	13	21
4	86	14	111
5	13	15	53
6	91	16	174
7	15	17	25
8	120	18	181
9	27	19	27
10	101	20	218

An easy conjecture is that for primes p, $\dim \mathrm{CA}(\rho_{D_p}) = p + 8$. Notice that the dimension is small for n odd. This is also observed in [BBST01], where the authors compute modular invariants for D_6, D_{10}, and D_{14}. They conjecture that for n an odd prime, and perhaps generally for n odd, the number of modular invariants does not increase drastically with n, while for n even the number of modular invariants can be quite large (they expect that D_8 may have more than 10^5).

Chapter 4

Modular data as a group invariant

In [CGR00], the authors raised the question of whether non-isomorphic groups can have 'the same' modular data. We provide some answers in this section.

4.1 Group invariants

By a *group invariant* we mean a property of a group that remains unchanged under group isomorphism. For example, group order, number and size of conjugacy classes, and the character table are group invariants.

Call a group invariant *complete* if groups are isomorphic if and only if their invariants are equal. Complete invariants for finite groups are quite complex. The main examples are k-characters and the group determinant (described in the next chapter). Most invariants are of course not complete. For example, both the dihedral and quaternion groups of order 8 have the same character tables, but are not isomorphic.

4.1.1 Modular data as an invariant

A permutation $P \in S_m$ has an associated permutation matrix $P \in M_m(\{0,1\})$. We will use the same symbol P to refer to both, and we make a few observations for a permutation matrix P and any matrix A:

- $P^{-1} = P^{\mathrm{Tr}}$ (Tr denotes the transpose)
- left multiplication of A by P permutes the rows of A by P
- right multiplication of A by P^{-1} permutes the columns of A by P

- conjugation of A by P is the simultaneous permutation of rows and columns

Matrices A and B are *permutation-congruent* if there exists a permutation matrix P such that $PAP^{-1} = B$. Equivalently, B is obtained from A by simultaneous permutation of rows and columns.

Isomorphic groups G and H have the same modular data by formulas (2.11) and (2.12). However, we fix orders on the bases $\mathcal{C}_{\mathcal{G}}$ and $\mathcal{C}_{\mathcal{H}}$ allowing us to write S and T and matrices. Most likely will we have $S^G \neq S^H$ and $T^G \neq T^H$ since the orderings on $\mathcal{C}_{\mathcal{G}}$ and $\mathcal{C}_{\mathcal{H}}$ might not correspond. However, an isomorphism between G and H gives rise to a permutation P describing the proper correspondence. Writing P as a permutation matrix this gives

$$PS^GP^{-1} = S^H \text{ and } PT^GP^{-1} = T^H$$

Definition 4.1 *We say that modular data for G and H is* weakly equivalent *if there exists a permutation matrix P such that $PS^GP^{-1} = S^H$ and $PT^GP^{-1} = T^H$.*

Notice that G and H with weakly-equivalent modular data induce equivalent representations of $\mathrm{SL}_2(\mathbb{Z})$. The converse, however, need not be true. We consider the matrices S and T to be group invariants, subject to the above definition of equivalence (the representation ρ is also an invariant). One then asks whether or not modular data is a complete invariant.

This question was asked in [CGR00], though they do not specify precisely what it should mean for modular data to be equivalent. Both [Cun05] and [BBST01] provide an answer. Both use the definition of 'weak equivalence' above. However, [Cun05] concludes that there are 2 groups of order 16 with equivalent modular data while [BBST01] states that these same groups do not have equivalent modular data.

We will investigate weak equivalence and see that [Cun05] is correct. Consequently, weak equivalence is not very interesting. We propose a more restrictive (and less naive) definition of equivalence and show that under this definition modular data determines the group for groups of order less than 128. This is one of the major original results of the thesis. Our definition involves placing some very natural restrictions on ordering $\mathcal{C}$.

4.1.2 Ordering on $\mathcal{C}$

Any ordering we place on $\mathcal{C}$ must of course be invariant under isomorphism. For fixed $g \in R$ call the set $\{(h, \chi) \in \mathcal{C} \mid h = g\}$ the *g-block* and denote it $(g, *)$. Place the following restrictions on ordering $\mathcal{C}$:

- Every g-block must appear contiguously.
- The e-block appears first.
- After the e-block, the blocks appear in order of ascending size.
- Within each g-block, the element $(g, 1)$ appears first.

The restrictions do not define a total order since we may permute blocks of the same size (except for $(e, *)$) and we may permute elements within a block (except for $(g, 1)$). We call such permutations *valid permutations.* They form a subgroup of S_m which we denote $\mathrm{Sym}(\mathcal{C})$. A *valid ordering* on $\mathcal{C}$ is any total order conforming to the above restrictions. $\mathrm{Sym}(\mathcal{C})$ maps a given valid order to every other valid order. Henceforth we assume $\mathcal{C}$ is endowed with a valid order.

The ordering restrictions define the the *block structure* of the basis. Let $(e, *)$, $(g_2, *)$,$(g_3, *)$,...,$(g_l, *)$ be the blocks of the basis, ordered as above. Then the block structure of $\mathcal{C}$ is the length l vector of sizes of the blocks,

$$\text{block structure} = (|(e,*)|, |(g_2,*)|, \dots, |(g_l,*)|)$$

Bases $\mathcal{C}_G$ and $\mathcal{C}_\mathcal{H}$ have equal block structure iff $\mathrm{Sym}(\mathcal{C}_G) = \mathrm{Sym}(\mathcal{C}_\mathcal{H})$ The block structure define a block-matrix structure on S and T.

4.1.3 Strong equivalence

Definition 4.2 *We say that modular data for G and H is* strongly equivalent *if there exist permutation matrices $P \in \mathrm{Sym}(\mathcal{C}_G)$ and $Q \in \mathrm{Sym}(\mathcal{C}_H)$ such that $PS^GP^{-1} = QS^HQ^{-1}$ and $PT^GP^{-1} = QT^HQ^{-1}$. Let 'equivalent modular data' mean 'strongly equivalent modular data'.*

We say that S^G and S^H are strongly-equivalent even $PT^GP^{-1} \neq QT^HQ^{-1}$ (similarly for T^G and T^H). Observe that Q is necessary for the definition of strong equivalence but not for weak equivalence. Strong equivalence gives the equation $(Q^{-1}P)S^G(Q^{-1}P)^{-1} = S^H$, but $Q^{-1}P$ need not be in $\mathrm{Sym}(\mathcal{C}_G)$ since G and H might have different block structures. Consequently, $(Q^{-1}P)S^G(Q^{-1}P)^{-1}$ might not be a valid expression of S^G since the ordering restrictions might be violated. For groups with the same block structures, we do have $Q^{-1}P \in \mathrm{Sym}(\mathcal{C}_G)$ so we can simplify the definition of strong equivalence to simply the existence of $P \in \mathrm{Sym}(\mathcal{C}_G)$.

This has huge implications in testing for equivalence. If G and H have different block structures, we (in theory) need to consider every $P \in \mathrm{Sym}(\mathcal{C}_G)$ paired with every $Q \in \mathrm{Sym}(\mathcal{C}_H)$. If they have the same block structure, we need only determine if P exists. Even so, there are many possibilities for P. Note however that for weak equivalence P could be any element of $S_{|\mathcal{C}|}$, but ordering restrictions on $\mathcal{C}$ dramatically reduce the possibilities for P in strong equivalence. We also feel it is important that the ordering restrictions are determined only by the basis and do not require significant computation to determine. One could, for example, insist that $\mathcal{C}$ be ordered such that the entries of T are in increasing order. This would, however, require one to compute T before ordering $\mathcal{C}$.

One further restriction we *could* place on the ordering is to order elements $(g, \chi_1), (g, \chi_2), \ldots$ within a block by the degrees of the characters χ_i. This ordering is easy to compute, but does not tend to impose much restriction as many of the characters have degree 1. It also means more difficulty in programming, and we will see later that it is not necessary.

4.1.4 An example: D_4

As an example, the dihedral group D_4 has (ordered) character basis

$$\begin{aligned}
\mathcal{C}_{D_4} \;=\; & ((e,1),(e,\psi_1),(e,\psi_2),(e,\psi_3),(e,\chi_1),\\
& (x,1),(x,\phi_{0,1}),(x,\phi_{1,0}),(x,\phi_{1,1}),\\
& (xy,1),(xy,\varphi_{0,1}),(xy,\varphi_{1,0}),(xy,\varphi_{1,1}),\\
& (y,1),(y,v_1),(y,v_2),(y,v_3),\\
& (y^2,1),(y^2,\psi_1),(y^2,\psi_2),(y^2,\psi_3),(y^2,\chi_1))
\end{aligned}$$

Refer to §2.4.2 for the definitions of the above characters. The block structure is $(5,4,4,4,5)$. If we write the basis in numerical indices as

$$((1,2,3,4,5),(6,7,8,9),(10,11,12,13),(14,15,16,17),(18,19,20,21,22))$$

we can see that a valid permutation is $(2,3)(6,10)(7,12)(8,11)(9,13)(19,20,21)$. Some invalid permutations are $(6,7)$, $(14,19,15,20,16,21,17,22,18)$, and $(1,18)(2,19)(3,20)(4,21)(5,22)$. The basis has size 22, but $|\mathrm{Sym}(\mathcal{C}_{D_4})| = 746496$ as compared with $|S_{22}| = 22! = 1124000727777607680000$. All but two of the characters have degree 1.

4.1.5 Connection with graph isomorphism

Solving our modular data equivalence problem means determining permutation-congruency of symmetric matrices (with some restrictions). This is essentially the *edge-coloured graph isomorphism* problem.

By an *edge-coloured graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, c)$ we mean a graph with vertex set $\mathcal{V}$, edge set $\mathcal{E}$, and a colouring of edges $c : \mathcal{E} \to \{\text{some set of labels}\}$. We can represent $\mathcal{G}$ as an adjacency matrix A where

$$A_{ij} = \begin{cases} \text{colour of edge } (i,j) & \text{when edge } (i,j) \text{ exists} \\ 0 & \text{when } (i,j) \text{ is not an edge} \end{cases}$$

An *edge-coloured graph isomorphism* between graphs $\mathcal{G}$ and $\mathcal{H}$ is a bijection $\phi : \mathcal{V}_{\mathcal{G}} \to \mathcal{V}_{\mathcal{H}}$ between vertex sets such that

- $(i,j) \in \mathcal{E}_{\mathcal{G}} \iff (\phi(i), \phi(j)) \in \mathcal{E}_{\mathcal{H}}$
- $c_{\mathcal{G}}(i,j) = c_{\mathcal{H}}(\phi(i), \phi(j))$

In terms of adjacency matrices, applying ϕ corresponds to a simultaneous row and column permutation. So an (edge-coloured) graph isomorphism is a permutation P such that $PA_{\mathcal{G}}P^{-1} = A_{\mathcal{H}}$.

Our modular data equivalence problem is essentially edge-coloured graph isomorphism. Consider the matrix S as the adjacency matrix for a graph, the entries of S being the "colours" of the edges (the actual numerical values are unimportant). We observe that since S is symmetric, the corresponding graph is undirected. Entries on the diagonal of S represent loops in the graph. T is diagonal, so the corresponding graph is just nodes with loops. S tends to have very few distinct entries, so the corresponding graph has few colours. For example, the S matrix for D_4 has 484 entries but only 6 distinct values. Considering S as an edge-coloured graph allows us to apply a powerful graph isomorphism algorithm to determine permutation-congruence, which we describe in §4.3. It is not useful for T since the corresponding graph has very little structure, but we will see that permutation-equivalence of T matrices is easy anyway.

4.2 Computational technique

4.2.1 Building modular data

We use the GAP computational discrete algebra system [GAP05a]. GAP contains a library of groups of 'small' order [GAP05b], which includes all groups of order less than 1024 (and actually much more). The library was computed by Besche, Eick, and O'Brien and is also used in the MAGMA system. Let (n, m) denote the m^{th} group of order n in the data base.

Building S and T matrices is straightforward in GAP. Functions for computing conjugacy classes, centralizer, character tables, etc. are all provided so we can easily build S and T using formulas (2.13) and (2.12). We found that building S via (2.13) was considerably faster than using (2.11). We also have the option of building S and T in the $\mathcal{C}$ basis by first computing in the $\mathcal{P}$ basis, then using the change-of-basis formula (Proposition 2.5) to change bases. Finding S and T relative to $\mathcal{P}$ is computationally easy. Computing the change-of-basis matrix is harder. For T, using $\mathcal{C}$ directly is far better since T is diagonal relative to $\mathcal{C}$. For S, we found that for smaller groups (order < 32), the direct computation in $\mathcal{C}$ was faster while for larger groups changing bases was faster. GAP code for both methods is given in Appendix A.1.

4.2.2 Ordering cyclotomics

The entries of S and T are cyclotomic numbers over $\mathbb{Q}$, and we will need to be able to order these. The GAP system defines a total order on cyclotomics, as follows. Define the *conductor* of a cyclotomic ζ to be the smallest integer m such that ζ is in $\mathbb{Q}(\xi_m)$. Then cyclotomics are ordered according to the following.

(a) Rationals are ordered as usual.

(b) Rationals are less than irrational cyclotomics.

(c) For cyclotomics with different conductors, the one with the smaller conductor is less.

(d) For cyclotomics with the same conductor m, GAP uses an ordered basis of $\mathbb{Q}(\xi_m)$ as a $\mathbb{Q}$-vector space, called the Zumbroich basis. This basis has size $[\mathbb{Q}(\xi_m) : \mathbb{Q}] = \phi(m)$ (Euler totient). Each cyclotomic then corresponds to a length $\phi(m)$ $\mathbb{Q}$-vector, and these vectors are ordered lexicographically.

4.2.3 Which groups should be tested?

Our goal is to show that there are no two groups with order less than or equal to 128 that have (strongly) equivalent modular data. By no means do we need to compare every pair of such groups. First, we note that the $(e,1),(e,1)$-entry of S^G is

$$S^G_{(e,1),(e,1)|} = \frac{1}{|G|} \sum_{g\in K_e, h\in K_e\cap C_G(e)} 1 = \frac{1}{|G|}$$

In the ordering of C, $(e,1)$ is always first so $|G|^{-1}$ is always the top-left entry of S. So groups of different orders do not have equivalent modular data. Groups with the same order can have different sized bases, so such groups do not have equivalent modular data. Many groups with the same order do have the same sized bases so these are the pairs that we need to test. For example, there are 2328 groups of order 128 (up to isomorphism), with 22 different sizes of bases ranging from 172 to 16384.

We will focus on showing that pairs of groups DO NOT have equivalent modular data, so it is sufficient to show that either T^G and T^H or S^G and S^H are not equivalent. However, we do have a few pairs which we show are weakly-equivalent. We are mostly interested in strong equivalence, though many of the methods we use apply to both cases. Note that not weakly equivalent implies not strongly equivalent.

4.2.4 T equivalence

Determining equivalence of T matrices is easy. First, we use the *multiset* test. By a *multiset* we mean a collection which may contain several copies of the same element. The multiset corresponding to the matrix T^G is just the collection of all its entries. Then T^G and T^H are permutation congruent if and only if the are equal as multisets. For strong equivalence, equality as multisets is not sufficient so we define a canonical form for T.

Since T is diagonal, we will think of it as a tuple, broken into blocks according to the block structure. The canonical form is as follows:

(a) Each block is sorted in ascending order, with the exception that the $(g,1)$ entry remains first.

(b) Blocks of the same size are compared lexicographically and sorted in ascending order, with the exception that the $(e,*)$ block must be first.

There is a valid permutation mapping T into its canonical form, and we observe that this form is unique. Consequently, equal canonical forms implies strong equivalence. For G and H with the same block structure, the converse is true. For different block structures, the converse is not true. For groups with different block structures and different canonical forms, we make no further attempts to show T inequivalence and hope that the S matrices are inequivalent.

We always test for T non-equivalence first since it is computationally easy, and only proceed to test S non-equivalence when the T matrices are equivalent. This is very significant as S matrices can be very time-consuming and in some cases infeasible to build. At order 128, several groups have bases of size 16384 but all of these have non-equivalent T matrices. Building S for these groups would be unfeasible and determining equivalence could be nearly impossible.

4.2.5 Easy tests for S non-equivalence

Given a pair S^G and S^H, we have a few easy tests that can show non-equivalence.

(a) *Diagonal invariant.* Notice that the diagonal of PSP^{-1} is the diagonal of S permuted by P. For weak equivalence, S matrices must the same diagonals as multisets. For strong equivalence, take the diagonal of S and order it the same way we defined the canonical form of T. This forms the 'diagonal invariant'. For S matrices with the same block structure, different invariants implies non-(strong)-equivalence.

(b) *First-rows invariant.* Since the $(g, 1)$ entry of each block appears first, valid permutations do not move the first row of each block (though entries in the row can be permuted). Then, as with the diagonal, we can make another invariant from the first rows, ordering each row in ascending order and comparing rows of the same size lexicographically. For S matrices with the same block structure, the first rows appear in the same places in the matrix. Therefore S matrices with the same block structure but different first-row invariants are not equivalent.

(c) *Multiset test.* As with T, we can make a multiset from the elements of S. If the multisets corresponding to S^G and S^H are not equal, S^G and S^H are not equivalent.

(d) *Characteristic polynomial test.* Permutation-congruent matrices must have the same characteristic polynomial,

$$\begin{aligned} \det(PS^GP^{-1} - \lambda I) &= \det(P(S^G - \lambda I)P^{-1}) \\ &= \det(P)\det(S^G - \lambda I)\det(P^{-1}) \\ &= \det(S^G - \lambda I) \end{aligned}$$

though the converse is not true. GAP can compute characteristic polynomials. This test is not overly useful on more difficult cases and is computationally a bit slow (but programming it is dead easy).

The diagonal and first-rows invariants are important since we can compute these invariants without computing all the entries of S. This can both save time and make comparisons of certain large S matrices feasible.

4.2.6 Modular data with different block structures

Some pairs of groups have the same T canonical form but different block structures. To compare the S matrix of these groups, we cannot use the diagonal and first-rows invariants. The multiset test is sufficient for many of these cases but not all (the characteristic polynomial test is applicable but does not seem to be very helpful in these cases). For these cases, we have two simple tests.

Observe that the first block is always the e-block, and its first row is the $(e, 1)$-row. Even though S^G and S^H might not have the same size e-blocks, some submatrix of the larger e-block must be permutation congruent with the smaller e-block. Further, some sub-row of the larger first row must be permutation congruent to the smaller first row. Consequently, we have two necessary conditions for equivalence:

- The e-blocks of S^G and S^H are equal as sets (not multisets).
- The first row of the larger e-block contains, as a multiset, the first row of the smaller e-block.

These conditions, though simple, sufficed in all the cases that we needed.

4.2.7 When the easy tests fail...

Once a pair of groups G and H passes all the easy tests, we apply a graph isomorphism algorithm. This algorithm computes a 'certificate' using *equitable partitions*.

It is essentially the algorithm described in §7.3.2 of [KS99], which computes certificates for simple graphs. We make the obvious extension to handle edge-coloured graphs with loops. The next section describes this algorithm in detail.

4.3 Graph certificate algorithm

A *certificate* for a graph $\mathcal{G}$ is a complete invariant. So it is a property $\mathrm{Cert}(\mathcal{G})$ such that graphs $\mathcal{G}$ and $\mathcal{H}$ are isomorphic if and only if $\mathrm{Cert}(\mathcal{G}) = \mathrm{Cert}(\mathcal{H})$. The point of a certificate is that we need only test *equality* of certificates to determine isomorphism of graphs. The downside of course is that certificates are difficult to compute.

Let $\mathcal{G}$ be an m-vertex edge-coloured graph with adjacency matrix $A = A_{\mathcal{G}}$. Define $\mathrm{Num}(A)$ as the length $m(m+1)/2$ tuple formed by reading down the columns of A, starting on the left-most column and stopping at the diagonal element in each column. Let $\Pi_{\mathcal{G}} \subset S_m$ be any set of permutations of the vertices of $\mathcal{G}$ determined by the structure of $\mathcal{G}$ (i.e. not dependent on the ordering of vertices in A). For $\pi \in S_m$ being a permutation of the vertices of $\mathcal{G}$, let A_π denote the adjacency matrix for the permuted vertices (ie. $A_\pi = \pi A \pi^{-1}$, treating π as a permutation matrix). Then we define the certificate for $\mathcal{G}$ as

$$\mathrm{Cert}(\mathcal{G}) = \min\{\mathrm{Num}(A_\pi) \mid \pi \in \Pi_{\mathcal{G}}\}$$

The fact that $\Pi_{\mathcal{G}}$ is determined only by the structure of $\mathcal{G}$ ensures that this is in fact a certificate.

The choice of $\Pi_{\mathcal{G}}$ is extremely important. If we use all of $\mathrm{Sym}(m)$, the computation of $\mathrm{Cert}(\mathcal{G})$ becomes an NP-complete problem. Indeed, in the case when $\mathcal{G}$ is a simple graph (no edge colours), a minimum value of $\mathrm{Num}(A)$ has the maximum number of leading zeros. This comes from a maximum size submatrix of zeros, the vertices of which define a maximum independent set (i.e. a set of vertices with no common edges). Finding a maximum independent set is known to be NP-complete (it is equivalent to finding a maximum clique in the complement of $\mathcal{G}$). Our choice for $\Pi_{\mathcal{G}}$ is based on *refining equitable partitions to discrete partitions.*

4.3.1 Equitable partitions

An *ordered partition* of the vertices of $\mathcal{G}$ is an m-tuple $P = (P_1, P_2, \ldots, P_m)$ where $\{P_1, P_2, \ldots, P_m\}$ forms a partition of $\mathcal{V}$. The P_i are called *blocks*. P is *discrete* if

$|P_i| = 1$ for all i. A discrete partition P defines a permutation π of the vertices and hence defines a value for $\mathrm{Num}(A_\pi) = \mathrm{Num}(A_P)$. The *unit partition* is the partition with only one block.

Let the edge colours of $\mathcal{G}$ form an ordered set $\mathcal{K} = \{k_1, k_2, \ldots, k_t\}$. For modular data, $\mathcal{K}$ is the ordered set of the entries which appear in the matrix (see §4.2.2 for how cyclotomics are ordered). For a vertex $v \in \mathcal{V}$ and a subset $\mathcal{S} \subset \mathcal{V}$ define the *neighbour tuple of v relative to $\mathcal{S}$* as

$$\mathrm{Nbr}(v, \mathcal{S}) = (m_1, m_2, \ldots, m_t)$$

where $m_i = |\{(v,x) \in \mathcal{E} \mid x \in \mathcal{S},\ c(v,x) = k_i\}|$. So $\mathrm{Nbr}(v, \mathcal{S})$ is an ordered count of the number of edges of each colour that go between v and $\mathcal{S}$. We say a partition P is *equitable* if for all i, j and for all u, $v \in P_i$,

$$\mathrm{Nbr}(u, P_j) = \mathrm{Nbr}(v, P_j)$$

A discrete partition is equitable.

4.3.2 Partition refinement

The ordered partition $Q = (Q_1, \ldots, Q_s)$ is a *refinement* of the ordered partition $P = (P_1, \ldots, P_t)$ if every Q_i is contained in some P_j and whenever $u \in P_{i_1}$, $v \in P_{i_2}$ with $i_1 < i_2$ we have $u \in Q_{j_1}$, $v \in Q_{j_2}$ with $j_1 < j_2$. We now describe an algorithm that refines a partition P to an equitable partition Q. GAP source code is given in Appendix A.2.

Algorithm 4.3 (Refining algorithm) *Given an ordered partition P of $\mathcal{G}$, refine it to an ordered equitable partition.*

```
begin
1. Let $S$ be a list that is a copy of $P$.
2. while $S$ is not empty
3.         Remove the last block $U$ from $S$
4.         for each block $B$ of $P$
5.             Compute $\mathrm{Nbr}(b, U)$ for each $b \in B$
6.             Partition $B$ into subsets, each subset containing $b$ with the same
                   $\mathrm{Nbr}(b, U)$ value
7.             Order the subsets by increasing $\mathrm{Nbr}(b, U)$ value
```

```
8.          Replace $B$ with the ordered subsets
9.          Append the ordered subsets to $S$
end
```

Proof. We need to show that Algorithm 4.3 produces an equitable partition. Suppose not, i.e. in the final partition P there are blocks P_i and P_j and $b_1, b_2 \in P_i$ with $\mathrm{Nbr}(b_1, P_j) \neq \mathrm{Nbr}(b_2, P_j)$. Now P_j must have at some point been in S since S is initialized to P [Line 1] and any new blocks are appended to S [Line 9]. So P_j was considered at Line 3 as U. But the *for* loop [Line 4] ensures that every block of P has constant $\mathrm{Nbr}(-, P_j)$ value, contradicting the existence of P_i.

□

The ordering of P is done at Line 7 and depends only on the structure of $\mathcal{G}$. Indeed, suppose we have an isomorphism $\phi : \mathcal{G} \to \mathcal{H}$ and a partition P of the vertices of $\mathcal{G}$. Then the partition of $\mathcal{H}$ that results from running Algorithm 4.3 on $\phi(P)$ is the same as the partition of $\mathcal{H}$ resulting from running the algorithm on P then applying ϕ to the result.

We have one improvement to this algorithm that we use in implementation. If we are considering a block U at Line 3, but we have already considered blocks that form a partition of U, then U need not be used. Indeed, U cannot cause any block B to be split up since the subsets forming the partition of U would already have done so. This happens fairly often: a block B will be appended to S, but before this block is considered it will be partitioned by another block and the sub-blocks of B are appended to S and considered before B.

4.3.3 Certificate algorithm

Algorithm 4.3 generally produces a partition that is not discrete. To get a discrete partition we can take the first non-unit (i.e. size greater than 1) block P_i, choose $p \in P_i$ and split P_i into $\{p\}$ and $P_i \setminus \{p\}$ and refine the resulting partition using the refining algorithm. If P is still not discrete we repeat (using the first non-unit block of the refined partition) until we have a discrete partition, which defines a permutation of the vertices. Define $\Pi_{\mathcal{G},P}$ as the permutations resulting from all such choices of p. Then $\mathrm{Cert}_P(\mathcal{G}) = \min\{\mathrm{Num}(A_\pi) \mid \pi \in \Pi_{\mathcal{G},P}\}$. The details are Algorithm 4.4 below. Code is provided in Appendix A.2. Note that the certificate depends on the initial choice of partition P. For strong equivalence we need a different choice for P, as we will see later.

Algorithm 4.4 *Given a partition P of the vertices of $\mathcal{G}$, compute $Cert_P(\mathcal{G})$.*

```
1. Refine $P$ to an equitable partition using Algorithm 4.3.
2. if $P$ is not a discrete partition
3.         Find the first block $P_i$ of $P$ with $|P_i| > 1$
4.         for each element $p \in P_i$
5.                 Call Algorithm 4.4 with input $(P_1, \ldots, P_{i-1}, \{p\}, P_i \setminus \{p\}, P_{i+1}, \ldots, P_m)$
                           and store the certificate.
6.         Return the least certificate found.
7. else
8.         Return $\mathrm{Num}(A_P)$.
```

4.3.4 Algorithm improvements

Algorithm 4.4 is a recursive backtracking algorithm. In Line 5, one element is split off from its block and the algorithm is called recursively. The new partition is refined to be equitable. Unfortunately, the refinement procedure might not change the partition very much and consequently the search tree for the algorithm can be extremely large. There are, however, two ways to prune the search tree (both given in [KS99]) and one further improvement (due to the author).

The first method is to prune by remembering the least certificate found so far. Suppose we have an equitable partition P, with P_i being the first non-unit block and $i > 1$. Then $P = (\{p_1\}, \{p_2\}, \ldots, \{p_{i-1}\}, P_i, \ldots, P_m)$, so the top left $(i-1) \times (i-1)$ submatrix of A_P is already fixed and the first $(i-1)i/2$ entries of $\mathrm{Num}(A_P)$ are determined. We can compare this initial portion of $\mathrm{Num}(A_P)$ with the corresponding initial section of the best certificate found so far. If the best certificate is less, then this partition P cannot produce a lesser certificate so we can abandon this branch of the search tree.

The second method is to use automorphisms of $\mathcal{G}$. We can collect automorphisms at Line 8 of Algorithm 4.4 if we record both the best certificate c_{best} and the permutation π_{best} that produced it. If at Line 8 we find that $\mathrm{Num}(A_P) = c_{best}$ then $A_\pi = A_{\pi_{best}}$ (only equal adjacency matrices define the same Num). Then

$A_{\pi_{best}\pi^{-1}} = A$, hence $\pi_{best}\pi^{-1}$ is an automorphism of $\mathcal{G}$. We store these automorphisms. This will of course only discover some (possibly none) of the automorphisms of $\mathcal{G}$.

We use these automorphisms at Line 5 of the certificate algorithm. Suppose there exists and automorphism π of $\mathcal{G}$ such that

$$\begin{aligned}\pi(P) &= (\{\pi(p_1)\}, \ldots, \{\pi(p_{i-1})\}, \{\pi(p)\}, \pi(P_i \setminus p), \pi(P_{i+1}), \ldots, \pi(P_m)) \\ &= (\{p_1\}, \ldots, \{p_{i-1}\}, q, \pi(P_i \setminus \{p\}), \pi(P_{i+1}), \ldots, \pi(P_m))\end{aligned}$$

where $q \in P$ is an element that has been considered earlier in the *for* loop. That is, suppose π fixes the first $i-1$ elements and replaces p with an element that has already been considered in that position. Then since π is an automorphism and q has been considered already, we know that this partition will yield the same certificate as the one with q in place of p, so we can abandon this branch of the search tree.

We do not know the entire automorphism group of $\mathcal{G}$, but we do know the subgroup generated by the automorphisms we have collected. We can search this subgroup for π and if it exists in the subgroup we can prune the search tree. GAP has the capacity to generate the subgroup and search it for the existence of π (using the GAP commands `Group` and `RepresentativeAction`).

There is also an improvement to the refining algorithm that we can use in some cases. Suppose in Algorithm 4.4 we find P_i with size 2, $P_i = \{p, q\}$. The algorithm will be splitting P_i into $\{p\}$ and $\{q\}$ and calling the refining algorithm. Then we need only initialize $S = (\{p\}, \{q\})$ at Line 1 of Algorithm 4.3, instead of $S = P$. Indeed, any block P_j with $j \neq i$ cannot cause any other such block to split during the refining step since P is equitable before breaking up P_i. Nor can P_j split the new blocks $\{p\}$ and $\{q\}$ as they are singletons. So the P_j are not needed in S.

4.3.5 Certificate algorithm and modular data

The certificate algorithm depends on the initial partition P. P must only depend on the structure of the graph (not the ordering of the vertices) otherwise isomorphic graphs could have different certificates. The obvious choice for P is the unit partition (all vertices in one block). This is what we use for weak equivalence as it allows any permutation. For strong equivalence, we would like a partition P so that matrices have the same certificate if and only if they are strongly-equivalent. This does not seem to be possible, but we can come close.

The ordering restrictions on the character basis C defines a partition P by making each g-block a partition block. The restrictions do not tell us how to order blocks of the same size so we do not have a unique ordered partition. Our compromise is to union all blocks of the same size into a single block (except that the e-block remains by itself) and use this as the starting partition P. The consequence of this choice is that two S matrices with the same certificate need not be (strongly) equivalent since the permutations defining the certificates need not be valid permutations, but S matrices with different certificates are not be equivalent. This negative result suffices for all the cases we tested. This choice of P speeds up the algorithm considerably compared to using the unit partition.

4.4 S and T simultaneously equivalent

Using the above methods we found no candidates for strong equivalence, but several cases had weakly-equivalent T and S matrices (independently). The challenge was then to determine if the the modular data was in fact weakly-equivalent, i.e. the existence of P giving simultaneous equivalence, $PT^GP^{-1} = T^H$ and $PS^GP^{-1} = S^H$.

4.4.1 Backtracking algorithm

We employed a straightforward backtracking algorithm. The algorithm recursively builds a discrete ordered partition $P = (\{x_1\}, \{x_2\}, \ldots, \{x_n\})$. Let $P_l = (\{x_1\}, \ldots, \{x_l\})$, $P(i) = x_i$, and S_{P_l,P_l} be the $l \times l$ matrix with i, j entry $S_{P(i),P(j)}$, $1 \leq i, j \leq l$.

At depth l, the first $l-1$ values of P are fixed and we choose x_l from the unused indices. For each choice of x_l, we check if $S^G_{P_l} = S^H_{P_l}$ and $T^G_{P_l} = T^H_{P_l}$. If so, recursively call the algorithm at depth $l+1$. Otherwise, continue to the next choice for x_l.

We remark that this is not a particularly efficient algorithm, though it served our purposes. We also found that the initial ordering of the matrices had a strong influence over the efficiency, in the case when G and H had weakly-equivalent data. The algorithm chooses values for x_l in increasing order, i.e. $x_1 = 1, x_2 = 2, \ldots$ is (without regard to pruning) the first choice for P. By default, we order the bases C_G and C_H by a valid order (§4.1.2). However, we found that this choice was particularly disadvantageous. By randomly choosing $p, q \in S_{|C|}$ and permuting S^G,T^G by p and S^H,T^H by q before running the algorithm we were able to find an equivalence in considerably less time. The ordering on C_G and C_H that produces the equivalence does not seem at all related to the block structure. This suggests that our definition of strong equivalence is a good one (for the purpose of making modular data a complete invariant).

4.5 Results

We give results for both weak equivalence and strong equivalence, though our focus is on strong equivalence.

4.5.1 Weak equivalence

The table below gives all pairs of groups with order less than 33 that have either permutation-congruent T matrices or permutation-congruent S matrices. For groups with equivalent T and S, we indicate whether T and S are simultaneously equivalent (i.e. whether the groups have weakly-equivalent modular data).

Groups	T equivalent	S equivalent	Weakly-equiv MD	$\|C\|$
$(16,3),(16,13)$	Yes	Yes	Yes	88
$(16,4),(16,12)$	Yes	No		88
$(32,2),(32,23)$	Yes	No		352
$(32,2),(32,24)$	Yes	Yes	Yes	352
$(32,2),(32,47)$	Yes	No		352
$(32,5),(32,38)$	Yes	Yes	Yes	352
$(32,7),(32,8)$	No	Yes		100
$(32,9),(32,40)$	Yes	No		184
$(32,9),(32,42)$	Yes	Yes	Yes	184
$(32,10),(32,13)$	Yes	No		184
$(32,10),(32,14)$	Yes	No		184
$(32,10),(32,41)$	Yes	No		184
$(32,13),(32,14)$	Yes	No		184
$(32,13),(32,41)$	Yes	No		184
$(32,14),(32,41)$	Yes	No		184
$(32,22),(32,48)$	Yes	Yes	Yes	352
$(32,23),(32,24)$	Yes	No		352
$(32,23),(32,47)$	Yes	No		352
$(32,24),(32,47)$	Yes	No		352
$(32,27),(32,49)$	Yes	Yes	Yes	184
$(32,29),(32,34)$	Yes	No		184
$(32,32),(32,35)$	Yes	No		184
$(32,40),(32,42)$	Yes	No		184

In particular, the pair (16,3) and (16,13) have equivalent modular data in the weak sense, as do several pairs at order 32 (the order 16 case is the one given in [Cun05]).

Theorem 4.5 *There exist non-isomorphic groups with weakly-equivalent modular data. The smallest-order examples are the order 16 groups with presentations* $G = \langle x, y \mid x^4 = y^4 = xyxy = 1,\ yx^3 = xy^3 \rangle$ *and* $H = \langle a, b, c \mid a^4 = b^2 = c^2 = cbca^2b = 1, bab = a, cac = a \rangle$.

These groups are not strongly-equivalent. In particular, the T matrices do not have the same canonical form, and the S matrices have different diagonal invariants. The ordering that we found giving weak equivalence does not seem to have any recognizable structure (and of course is is not a valid ordering in the sense of §4.1.2).

Corollary 4.6 *For n divisible by 16 there exist groups of order n with weakly-equivalent modular data.*

Proof. Let G and H be the order 16 groups above, with $PS^GP^{-1} = S^H$ and $PT^GP^{-1} = T^H$. Then $G \times C_m$ and $H \times C_m$ have weakly-equivalent modular data. Indeed, by 3.10 we have $S^{G\times C_m} = S^G \otimes S^{C_m}$ and $T^{H\times C_m} = T^H \otimes T^{C_m}$. Writing $S^G \otimes S^{C_m}$ as an $m \times m$ block matrix and applying P to the blocks gives the necessary equivalence.

□

We also observe that the groups (32,7) and (32,8) have different T but equivalent S matrices. This is a bit unexpected considering that T is much simpler than S.

4.5.2 Strong equivalence

For strong equivalence, we were able to prove the following theorem. It is one of the most important new results of the thesis.

Theorem 4.7 *There is no pair of groups with order 127 or less having strongly-equivalent modular data.*

We were also able to show that most groups of order 128 do not have strongly-equivalent modular data. However, we were left with 528 pairs for which we were unable to determine equivalence or non-equivalence.

There are 3596 groups with order less than or equal to 128. Of these, 2328 have order 128. The number of pairs with the same group order and same size character bases is around 353000. As mentioned earlier, most pairs have inequivalent T matrices: only 4371 pairs have the equivalent T matrices. Another 2502 pairs have T matrices for which we could not determine equivalence or inequivalence (the groups have different block structure and the T matrices, in the canonical form, are not equal). Consequently we had 6873 pairs for which the S matrices needed to be compared. The 'easy tests' are sufficient for most of these cases, eliminating 6322 pairs and leaving 551.

Of these remaining 551 pairs, we use the certificate algorithm to show that 23 of them have inequivalent S matrices. Of these 23, 9 occur at order 64, 1 at order 100, and the remaining 13 at order 128.

For the remaining 528 pairs, we were unable to determine their status. These all occur at order 128, most of them with $|\mathcal{C}| = 928$ and the largest with $|\mathcal{C}| = 2944$. In many of these cases the S matrices were simply too time-consuming to build, so we could not apply all the tests (the first-rows and diagonal invariants were essential in eliminating many large cases, since we avoided building the complete S matrix). In the other cases the certificate algorithm did not terminate in reasonable time.

We found 4371 pairs with strongly-equivalent T matrices. The smallest example is (16,4) and (16,12). The next smallest examples are at order 32: the pair (32,9) and (32,40), the pair (32,32) and (32,35), the triple (32,2),(32,23),(32,47) and the four groups (32,10),(32,13),(32,14),(32,41). Also, we do not know if the T matrices of (32,27) and (32,49) are strongly-equivalent (their S matrices are not strongly-equivalent though).

There are groups with strongly-equivalent S matrices. The smallest example is the order 32 groups (32,7) and (32,8). Their T matrices are not equal as multisets.

4.5.3 Program correctness

The GAP code for this study is about 1700 lines. We test this code in order to give confidence in the results. We are, however, reliant on the correctness of the GAP system and of the small groups library. We run two major tests.

(a) *S and T formula test.* We test the formulas for S and T by building S and T using the $\mathcal{C}$ basis and comparing it with the matrices built using the $\mathcal{P}$ basis and changing basis. We also test S built from (2.11) versus (2.13). All groups up to order 43 were tested successfully.

(b) *Certificate test.* To test the certificate algorithm, we build S and compute its certificate. Then permute $\mathcal{C}$ by a random valid permutation and build S from the reordered basis. Compute the certificate for the new S and ensure that it is equal to the original certificate. Since we are choosing valid permutations at random, we repeat several times (five) with different random valid permutations. We perform the test successfully for all groups up to and including order 44.

Chapter 5

k-characters and modular data

In the preceding chapter we saw that under the strong equivalence a group's modular data determines the group for small orders. Ideally, we would either prove that modular data is a complete invariant, or find two groups with (strongly) equivalent modular data. We have been unable to do either. We suspect that the modular data is not a complete invariant, but we will see in §5.3 some of the group properties that *are* determined by modular data. In §5.1 we review two properties that are complete invariants, the *group determinant* and *k-characters*. We will also look at *2-characters*, which are an invariant but do not determine the group, and see how the group properties determined by the 2-characters compare to those determined by modular data. Finally, we give a new result giving groups with the same 2-characters but different modular data.

5.1 The group determinant and k-characters

Group characters of non-Abelian groups were introduced by Frobenius to study the factorization of the *group determinant*. Let G be a finite group, and $\{x_g \mid g \in G\}$ a set of independent commuting variables. Let X_G be a matrix indexed by G, with the (g, h)-entry being $x_{gh^{-1}}$. Then the group determinant is $\Theta_G = \det(X_G)$. The indexing set G is not ordered, but every choice of order gives the same determinant since $\det(P\Theta_G P^{-1}) = \det(P)\det(\Theta_G)\det(P^{-1}) = \det(\Theta_G)$ for every permutation matrix P.

The group determinant is a complete invariant for finite groups, meaning G and H are isomorphic if and only if Θ_G and Θ_H are the same. This was first proved by Formanek and Sibley in 1991 [FS91], using maps between group algebras. A proof by Mansfield in 1992 is shorter and reconstructs the group multiplication directly from the group determinant by examining terms of the form $x_e^{n-2}x_g x_h$ and $x_e^{n-3}x_g x_h x_k$ [Man92].

We must be precise in what it means for two groups to have 'the same' determinant. For groups G and H, Θ_G is in the polynomial ring $\mathbb{Z}[x_g]$ while Θ_H is in $\mathbb{Z}[x_h]$. A bijection $\phi : G \to H$ induces a ring isomorphism

$$\begin{aligned} \hat{\phi} : \mathbb{Z}[x_g] &\longrightarrow \mathbb{Z}[x_h] \\ x_g &\longmapsto x_{\phi(g)} \end{aligned}$$

We say that G and H have the same determinant if such a bijection ϕ exists with $\hat{\phi}(\Theta_G) = \Theta_H$. For groups of order n, there are $n!$ choices for ϕ. Notice that this is essentially the same combinatorial problem that we had with the character basis $\mathcal{C}$ of modular data in Chapter 4. In that case we were able to give some order to $\mathcal{C}$ in a obvious way and without doing any computation. It is not clear whether we could place some order on G in a similar way. We could, for example, insist that the elements of G be collected together according to orders or conjugacy classes, but this might require extra computation. Consequently, it seems that comparing two group determinants is very difficult.

Of course, in many cases it will be easy to show that two group determinants are not the same, just as we had many easy cases for modular data. A simple test is to form the multiset consisting of the coefficients of Θ_G. If the multiset for Θ_G is not equal to that of Θ_H then Θ_G and Θ_H are not the same. It is very unlikely that this invariant actually determines Θ_G (i.e. does not determine G). But it is fun to check that it holds for some small order cases. We use GAP to prove the following (see Appendix A.4 for the GAP code):

Proposition 5.1 *The multiset consisting of the coefficients of the group determinant is unique for groups of order less than 10.*

The reason the result is given for such small order is that the group determinant is very slow to compute, since it is the computation of a *symbolic determinant* (i.e. the entries are the symbols x_g rather than numbers). Modular data by comparison is much easier to compute. It is not surprising that the group determinant is difficult to compute and compare since it is a complete group invariant.

5.1.1 k-characters

To factor the group determinant Frobenius introduced functions $\chi^{(k)} : G^k \longrightarrow \mathbb{C}$, where χ is an irreducible character of G and $k \in \mathbb{Z}_{\geq 1}$ called *k-characters.* (see [Fro68] and the summary in [Joh91]). These can be defined recursively by

$$\begin{aligned}
\chi^1(g_1) &= \chi(g_1) \\
\chi^k(g_1, g_2, \ldots, g_k) &= \chi(g_1)\chi^{k-1}(g_2, g_3, \ldots, g_k) \\
&\quad -\chi^{k-1}(g_1g_2, g_3, \ldots, g_k) - \ldots - \chi^{k-1}(g_2, g_3, \ldots, g_1g_k)
\end{aligned}$$

The 2-character is explicitly given by

$$\chi^{(2)}(g,h) = \chi(g)\chi(h) - \chi(gh)$$

and the 3-character is

$$\begin{aligned}
\chi^{(3)}(g,h,k) &= \chi(g)\chi(h)\chi(k) - \chi(g)\chi(hk) - \chi(h)\chi(gk) - \chi(k)\chi(gh) \\
&\quad +\chi(ghk) + \chi(gkh)
\end{aligned}$$

Remark 5.2 *To compute the above expression, we need to know that a character value $\chi(g_1g_2 \ldots g_k)$ depends only on the cyclic order of the factors $g_1, g_2, \ldots, g_k$. This is a consequence of the fact that characters are conjugacy class functions, so for example*

$$\chi(g_1g_2 \cdots g_k) = \chi(g_kg_1g_2 \cdots g_kg_k^{-1}) = \chi(g_kg_1 \cdots g_{k-1})$$

The definition of k-characters immediately yields the following proposition.

Proposition 5.3 *Let* $\deg(\chi) = m$.

(a) For $k > m$ then $\chi^{(k)} = 0$.

(b) The m-character, together with $\deg(\chi)$*, determines the k-characters for $k < m$.*

Proof. Letting $g_1 = e$ we get

$$\begin{aligned}
&\chi^{(k)}(e, g_2, \ldots, g_k) \\
&= \chi(e)\chi^{(k-1)}(g_2, \ldots, g_k) - \chi^{(k-1)}(g_2, \ldots, g_k) - \cdots - \chi^{(k-1)}(g_2, \ldots, g_k) \\
&= (\deg(\chi) - k + 1))\chi^{(k-1)}(g_2, \ldots, g_k)
\end{aligned}$$

Consequently $\chi^{(l+1)} = 0$ and by the recursion all higher k-characters are zero. If we know $\deg(\chi)$ we can compute χ^k for $k < m$.

□

Frobenius obtains the following result, showing how k-characters appear in the group determinant.

Proposition 5.4 *Let Θ be the group determinant of G. Then the number of irreducible factors of Θ is equal to the number of irreducible characters of G and each irreducible factor θ of Θ corresponds to an irreducible character χ_θ of G. The degree k of θ is equal to the degree of χ_θ, and θ is given by the formula*

$$\theta = \frac{1}{k!} \sum_{(g_1,g_2,\ldots,g_k)\in G^k} \chi_\theta^{(k)}(g_1, g_2, \ldots, g_k) x_{g_1} x_{g_2} \cdots x_{g_k}$$

5.2 Group properties determined by k-characters

In [Bra63], Brauer posed the question of what information is needed in addition to the character table to form a complete invariant for finite groups. He proposed additional conditions as follows. Let G have irreducible characters χ_i and conjugacy classes K_j. For a conjugacy class K of G and integer m, define $K^{[m]} = \{x^m \mid x \in K\}$. Observe that $K^{[m]}$ is a conjugacy class of G. Indeed, any x^m and y^m in $K^{[m]}$ are conjugate since $y = gxg^{-1}$ for some $g \in G$ hence $y^m = gx^m g^{-1}$. $K^{[m]}$ is closed under conjugation since $gx^m g^{-1} = (gxg^{-1})^m$. Now assume G and H have isomorphic character tables via $\chi_i(K_j) = (\pi\chi_i)(\sigma K_j)$. Then Brauer's additional condition is that $\sigma(K_j^{[m]}) = (\sigma K_j)^{[m]}$ for all i, m. Groups satisfying these conditions are known as a *Brauer pair.* Brauer asked whether such a pair must be isomorphic. The answer is no, first shown in [Dad64].

Proposition 5.4 shows that knowledge of all the k-characters of a group is sufficient to compute the group determinant, and therefore to determine the group. This gives an answer to Brauer's question. In fact, less information is needed. Hoehnke and Johnson showed in [HJ95] that knowledge of the 3-characters (which, as we know, gives the 1- and 2-characters) is sufficient for the group determinant. Consequently the 3-characters are a complete group invariant (and is a much simpler invariant than the group determinant).

This of course raised the question of whether or not the 2-characters determine the group. This was shown to be false in [JS93], where certain Brauer pairs are shown to have the same 2-characters. The smallest groups with the same 2-characters are the two non-Abelian groups of order 27, proved in [JS95]. The definition of G and H

having 'the same 2-characters' is the existence of a bijection π between irreducible characters of G and H and a bijection $\tau : G \longrightarrow H$ such that

$$\begin{aligned} \chi_i(g) &= (\pi\chi_i)(\tau(g)) \\ \chi_i^{(2)}(g,h) &= (\pi\chi_i)^{(2)}(\tau(g),\tau(h)) \end{aligned}$$

Since 2-characters do not determine the group, [JMS00] found group properties that are determined by the 2-characters. Define the *weak Cayley table* of G as the table with rows and columns indexed by G and whose (g,h)-entry is the conjugacy class of gh. Then [JMS00] shows that the weak Cayley table contains the same information as the 2-characters:

Proposition 5.5 *If the irreducible 1- and 2-characters of G are known, then the weak Cayley table can be constructed. Conversely, if the weak Cayley table is known then the irreducible 1- and 2-characters can be computed.*

By examining the values $\chi^{(2)}(g,g)$, [JMS00] shows that the following properties are determined by the 2-characters.

Proposition 5.6 *Given the 2-characters of G corresponding to irreducible 1-characters, the following are determined:*

(a) For each $z \in Z(G)$, the set $\{g \in G \mid g^2 = z\}$.

(b) For each conjugacy class K, the class $K^{[2]}$.

(c) The set of elements whose order is a power of 2, together with the order. In particular, the involutions (order 2 elements) of G.

(d) The Frobenius-Schur indicator of G.

5.3 What group properties are determined by modular data?

In Chapter 4, we saw that modular data determines the group for orders less than 128. We do not know if modular data determines the group for all orders, but in this section give properties of a group that are determined by modular data.

Proposition 5.7 *Given S and T in the character basis, ordered according to §4.1.2, the following information about G is known.*

(a) Order of G.

(b) Exponent of G.

(c) Character table of G.

(d) Block structure of $\mathcal{C}_G$.

(e) Sizes of the conjugacy classes and centralizers of G.

(f) Whether or not G is Abelian. If G is Abelian, we can determine G.

Proof. The order of G is $S^{-1}_{(e,1),(e,1)}$, which is the top-left entry according to the basis ordering. The exponent of G is the order of T, as we saw in Proposition 2.10. To determine the character table, we start by applying formula (2.14) to compute the ratio

$$\frac{S_{(a,\psi),(e,\chi)}}{S_{(a,\psi),(e,1)}} = \frac{\overline{\chi(a)\psi(e)}}{|C(a)|} \cdot \frac{|C(a)|}{\overline{1(a)}\,\overline{\psi(a)}} = \overline{\chi(a)}$$

That is, for a row (a, ψ), $\chi(a)$ is the conjugate of the (e, χ) column entry divided by the first column entry. Let $\chi_1, \chi_2, \ldots, \chi_k$ be the irreducible characters of G. We compute $\chi_1(e), \chi_2(e), \ldots$ using the first row. We reach the last irreducible character once $\sum_{i=1} \chi_i(e)^2 = |G|$ (Theorem 1.8), and we now also know the size of the e-block. Then for any row (a, ψ), we use the first k columns to get the values of all χ_i on the conjugacy class K_a. Since the irreducible characters are linearly independent class functions, two rows correspond to different conjugacy classes if and only if the character values are different. This gives the character table and also identifies each g-block, so we know the block structure of $\mathcal{C}_G$.

For the sizes of the centralizers we observe that the first entry in an $(a, 1)$ row is

$$\begin{aligned} S_{(a,1),(e,1)} &= \frac{1}{|C_G(a)||C_G(e)|} \sum_{g \in G(a,e)} \overline{1(geg^{-1})1(g^{-1}ag)} \\ &= \frac{1}{|C_G(a)||G|} \sum_{g \in G} 1 \\ &= \frac{1}{|C_G(a)|} \end{aligned}$$

and we can identify the $(a, 1)$ row since we know the block structure. The size of the conjugacy class is given by Lemma 1.1.

G is Abelian if and only if it has $|G|$ conjugacy classes, if and only if $|\mathcal{C}| = |G|^2$. Since we know $|G|$ we can determine if G is Abelian. An Abelian group is determined by its character table.

□

Next we consider what group information is contained in the modular data relative to the permutation basis. For fixed a, define the *a-block* as $(a, *) = \{(g, h) \in \mathcal{P} \mid g = a\}$. Place the following ordering restrictions on $\mathcal{P}$:

(a) Every a-block must appear contiguously.

(b) The e-block appears first.

(c) After the e-block, the blocks appear in order of ascending size.

(d) Within each a-block, the element (a, e) appears first.

Notice that these ordering restrictions correspond with those we placed on $\mathcal{C}$, so both bases have the same block structure.

Proposition 5.8 *Suppose we are given S and T with respect to the basis $\mathcal{P}$, ordered as above. Then we know the following information.*

(a) The block structure of the basis.

(b) The order of the elements in each conjugacy class.

Proof. To identify the e-block, we have $[\![e, g]\!].t = [\![e, g]\!]$, hence T has a 1 on the diagonal for each $[\![e, h]\!]$. The next block starts with $[\![g, e]\!].t = [\![g, g]\!] \neq [\![g, e]\!]$, so the first non-diagonal entry of T indicates the start of the next block. Now examine S. Since $[\![g, e]\!].s = [\![e, g^{-1}]\!]$, the first column of each block is identified by an entry in one of the e-block rows. For the remaining elements of the block (i.e. $h \neq e$) we have $[\![g, h]\!].s = [\![h, g^{-1}]\!]$, so the entry in S is outside the e-block rows.

For the order of each conjugacy class, find the column of the first entry $[\![g, e]\!]$ of its block, as above. Since $[\![g, e]\!].t^l s^3 = [\![g, e]\!].\left(\begin{smallmatrix} l & 1 \\ -1 & 0 \end{smallmatrix}\right) = [\![g^l, g]\!]$, the smallest l such that $g^l = e$ is the smallest l such that the entry in the $[\![g, e]\!]$ column of $T^l S^3$ is in the e-block rows.

□

If we have modular data in both bases, we know the orders of all the elements of G:

Proposition 5.9 *Suppose we know S and T with respect to both the character and permutation bases. Then in addition to the information from Propositions 5.7 and 5.8 we know the list of orders of the elements of G.*

Proof. The character basis gives the size of each conjugacy class, and the permutation basis gives the order of elements in each class.

□

We saw earlier that the 2-characters determined the number of elements with order a power of 2, along with the order. Proposition 5.9 is stronger, but still is not a very strong condition. The smallest groups with the same orders of elements are at order 16, for example $C_4 \times C_4$ and the group with presentation $G = \langle x,\ y \mid x^4 = y^4 = e,\ xy = yx^3\rangle$. These groups both have elements with orders 1,2,2,2,4,4,4,4,4,4,4,4,4,4,4,4.

Proposition 2.5 gives the formula to change bases between $\mathcal{C}$ and $\mathcal{P}$. The formula requires the knowledge of the character tables of all the centralizers $C_G(a)$. It is not clear whether this information can be obtained from S and T (in either basis). It turns out that there are groups with inequivalent modular data that have the same character tables of centralizers.

For a character basis $\mathcal{C}_G$, consider the multiset consisting of character tables of the centralizers $C_G(a)$, $a \in R$. Say that two character bases are *character table equivalent* if these multisets are equal, with isomorphic character tables counting as equal. The following proposition is new.

Proposition 5.10 *The smallest groups with character table equivalent bases are at order 64. These groups have inequivalent modular data.*

Proof. We content ourselves with a computer-assisted proof, using GAP. Computing the centralizers and their character tables is straightforward. Testing character table is also straightforward in GAP: the `TransformingPermutations` function determines the existence (and finds) permutation matrices P and Q such that for given matrices A and B, $PAQ = B$. Notice that this means rows and columns may be permuted differently, as is allowed in character table equivalence (unfortunately, this also prevents us from using `TransformingPermutations` to test modular data equivalence since we need $P = Q$ in that case). Code is provided in Appendix A.5. We find that there are 6 pairs at order 64 with character-equivalent bases, given by the GAP library numbers (64,13) with (64,14), (64,70) with (64,72), (64,74) with (64,80), (64,142) with (64,157), (64,143) with (64,158), and (64,175) with (64,181). Checking with our results from Chapter 4, we find, not surprisingly, that all these pairs have equivalent T matrices, and have S matrices that pass all of the 'easy tests' (e.g.. equal as multisets). The pairs all have different certificates though, so their modular data is inequivalent (in the weak, and therefore the strong, sense).

□

5.4 Groups with the same 2-characters but different modular data

We will show that there exist groups with the same 2-characters, but inequivalent modular data. It is well-known that for p and odd prime, there are 2 non-Abelian groups of order p^3. These groups have the following presentations, given in [You93].

$$\begin{aligned} P &= \langle a, x, y \mid a^p = y^p = 1,\ x^p = a,\ yx = axy,\ ax = xa,\ ay = ya \rangle \\ Q &= \langle a,\ b,\ c \mid a^p = b^p = c^p = 1,\ cb = abc,\ ab = ba,\ ac = ca \rangle \end{aligned}$$

Johnson and Sehgal show that these groups have the same 2-characters [JS95]. We show that their modular data is non-equivalent. This result is new.

Proposition 5.11 *The groups P and Q above have non-equivalent modular data. In particular, their T matrices are not equal as multisets.*

Proof. P has exponent p^2 while Q has exponent p. Indeed, the exponent divides the order of the group so the only possibilities are p and p^2 (not p^3 since the groups are non-Abelian). In P, x has order p^2 hence P has exponent p^2. In Q, elements can be written as $a^i b^j c^k$ where $0 \leq i, j, k \leq p-1$. The pth power of an element is

$$\begin{aligned} (a^i b^j c^k)^p &= a^{pi + (\sum_{t=1}^{p-1} t) jk} b^{pj} c^{pk} \\ &= a^{pi + \frac{p(p-1)}{2} jk} \\ &= e \end{aligned}$$

hence Q has exponent p.

Since the exponent of the group is the order of T, the groups P and Q have T matrices that are inequivalent as multisets. Consequently their modular data is non-equivalent.

□

Chapter 6

Concluding remarks

In this thesis we examined some properties of finite group modular data that had until now been poorly-explored. Our most important original results were the following:

- S and T determine the group for groups of order less than 128, under a new definition of equivalence.
- the dimension of the centralizer algebra of $\mathbb{Z}_n$ modular data
- the decomposition of $\mathbb{Z}_p$ (p a prime) modular data into irreducible $\mathrm{SL}_2(\mathbb{Z}_p)$ representations
- existence of groups with the same 2-characters but different modular data

Each of these results has raised further research questions.

Modular data as a group invariant

We do not know whether S and T determine the group, under the new definition of equivalence. We showed that most groups of order 128 had inequivalent S and T, but were left with 528 pairs whose status we could not determine. Our current methods, with minor improvements and more computing time, can probably deal with these 528 pairs. These pairs are certainly a place to look for a counter-example. The fact that there are groups with strongly-equivalent S matrices (the smallest pair being at order 32) may suggest that we will find a counter-example. However, if there is no counter-example at order 128, it may be worth trying to prove that modular data determines the group. The paper [Dav00] may be useful in this respect.

We don't have a guess as to what the answer is. The permutations giving the weak-equivalence in the order 16 and 32 cases do not respect the block structure of

$\mathcal{C}$. We were surprised by this, as we thought that making the bases $\mathcal{C}_G$ and $\mathcal{C}_H$ correspond would make it easier to find a permutation giving equivalence. Proposition 5.10, showing that groups can have 'character table equivalent bases' (the same character tables of centralizers) but still have inequivalent modular data is also telling. It suggests that the set $G(a,b)$ is particularly important in the definition of S, and is worth examining more closely.

2-characters

Though we proved that groups may have the same 2-characters but different modular data, we have further questions about how 2-characters may be related to modular data. We can get the character table from S and T. Can we get the 2-character table? If modular data does not determine the group, are there groups with the same S and T but different 2-characters? We also saw that the 2-characters determine the Frobenius-Schur indicator, but did not see anything similar for modular data. In CFT there is an analogue of the Frobenius-Schur indicator, defined in terms of entries from S and T [Ban97]. Perhaps it provides a way to recover the character-theory indicator?

On a side note, 2-characters and k-characters are interesting in themselves. Group characters (1-characters) span the space of class functions on G. In [Joh91], Johnson defines 'extended 2-characters' (2-character plus some other combinations of 1-characters), which are (pairwise) orthogonal but do not span the space of '2-class functions'. Is there a better generalization of group characters that do span the space of k-class functions? We remark that k-characters are built from 1-characters: they are not really a generalization of 1-characters but rather way of extracting more information from 1-characters.

Centralizer algebra and decomposing ρ_G

Since $\mathrm{SL}_2(\mathbf{Z}_p)$ has a nice representation theory, we expect that for any group with exponent p we should be able to decompose ρ_G as we did for $\mathbf{Z}_p$. For $\mathrm{SL}_2(\mathbf{Z}_{p^n})$, the representation theory is more complicated ($\mathbf{Z}_{p^n}$ is not a field). The irreducible representations of $\mathrm{SL}_2(\mathbf{Z}_{p^n})$ are known and are given in [NW76] (and in English in [Eho95]), though they are not given directly. Still, the decomposition for $\mathbf{Z}_n$ should certainly be possible.

As we remarked in Chapter 3, decomposing ρ_G into irreducibles gives the structure of the centralizer algebra, but to get the centralizer algebra itself (relative to the C basis) we need the change-of-basis matrix from C to the basis in which ρ_G is block-diagonal. It is not obvious how to find it. But knowing the change-of-basis gives the centralizer algebra and is a major step towards knowing the modular invariants. This approach to finding modular invariants for finite group data is worth further research.

For the dimension of the centralizer algebra, we expect that staring long enough at the data we presented for D_n will yield a nice formula, with a similar proof to the $\mathbb{Z}_n$ case. We should also find the dimension of the centralizer algebra for other group families, with the symmetric and alternating groups being a major goal.

Group properties in modular data

We saw that modular data of a direct product of groups is the tensor product, but we do not know if there is a similar relation for the semi-direct product. This is worth exploring further. In general, we want to know how group-theoretic properties and constructions are reflected in modular data. We understand modular data for abelian groups. If G is simple (nilpotent, solvable, etc.) what can we say about its modular data?

CFT on higher-genus surfaces

Recall from §2.3.1 that modular data in the CFT context arose when the CFT had the torus as its space-time. More generally, we are interested in CFT living on any world-sheet of any number of strings, i.e. CFT on surfaces of genus g with n punctures (each puncture corresponds to string, and the genus tells how many times a string splits apart and later reforms). Higher-genus surfaces can be built from lower-genus surfaces. For example, take two tori, each with one puncture, and 'glue them together' at the puncture. The result is a genus 2 zero puncture surface ('double torus'). In this way the torus and its corresponding modular data is important as a building block of more general cases (incidentally, modular data also corresponds in a way to the torus with one puncture).

In the general genus g with n punctures case, the partition function $\mathrm{ch}_A(\tau)$ becomes a *correlation function*, with τ living in the moduli space of the surface. Instead of $\mathrm{SL}_2(\mathbb{Z})$, we get the *mapping class group* of the surface. The role of the characters is played by *conformal blocks*, and these yield a representation of the mapping class group just as the characters give a representation of $\mathrm{SL}_2(\mathbf{Z})$ (which is the mapping class group of the torus). These representations have yet to be studied carefully and is an avenue for future research.

The action of the braid group B_n on G^{n-1} will probably be important in the representations of the higher-genus mapping class groups. The genus g with n punctures mapping class group is built from the genus g zero punctures group and a braid group. The exact relationship is described in [Bir75].

Appendix A

GAP source code

Select GAP source code is given below. Lines beginning with '#' are comments. Source code for some simple functions is omitted, though in such cases a short description of the function is given in comments. Any other functions that are not documented are standard GAP functions. Documentation for GAP is available from the GAP website, `http://www.gap-system.org/`.

A.1 Building modular data

The following code is used to build the bases $\mathcal{C}$ and $\mathcal{P}$ and the matrices S and T.

```
#--------------CharBasis----------------------
# Given a finite group G, return the character
# basis of G.  The basis is ordered as follows:
#   1. The (e,*)-block is first
#   2. Every (g,*)-block appears contiguously
#   3. The (g,1) entry is first in each block
# The basis is listed as 4-tuples [g,c,i,j] where
#   g=element of R
#   c=character of C(g)
#   i=block number
#   j=0 if c is trivial character, 1 otherwise
CharBasis:=function(G)
local R,CT,CTg,I,C,Basis,c,g,i,D,p;
CT:=CharacterTable(G);
R:=List(ConjugacyClasses(CT), Representative);
Basis:=[];
i:=0;
for g in R do
    i:=i+1;
    C:=Centralizer(G,g);
    CTg:=CharacterTable(C);
    I:=Irr(CTg);
    # Run through I to find the trivial character and put it first.
    for c in I do
        # Identity(c) returns the identity character of underlying character table
        if c=Identity(c) then
```

```
            Add(Basis,[g,c,i,1]);
        fi;
    od;
    # Run through the remaining characters of I
    for c in I do
        if c<>Identity(c) then
            Add(Basis,[g,c,i,0]);
        fi;
    od;
od;
# OrderBasis sort the basis blocks by size, but with (e,*) first
return OrderBasis(Basis);
end;

#------------PermBasis---------------------------
# Given G return the permutation basis of G.  The
# basis is given as 4-tuples [g,h,i,j] as follows:
#    g=reps of conjugacy classes of G
#    h=reps of conjugacy classes of C(g)
#    i=block number
#    j=1 if h=e, 0 otherwise
# The basis is ordered as folows:
#    1. Each (g,*)-block appears contiguously
#    2. The (e,*)-block is first
#    3. (g,e) is first in each block
PermBasis:=function(G)
local R,Basis,C,Rc,g,h,i,Blocks,perm;
R:=List(ConjugacyClasses(G),Representative);
Basis:=[];
i:=0;
for g in R do
    i:=i+1;
    C:=Centralizer(G,g);
    Rc:=List(ConjugacyClasses(C),Representative);
    for h in Rc do
        if h=Identity(h) then
            Add(Basis,[g,h,i,1]);
        fi;
    od;
    for h in Rc do
        if h<>Identity(h) then
            Add(Basis,[g,h,i,0]);
        fi;
    od;
    Rc:=[];
od;
# OrderBasis sort the basis blocks by size, but with (e,*) first
return OrderBasis(Basis);
end;

#------------------CharTlistb----------------------
# Return T as a list, relative to the character basis
# (remember T is a diagonal matrix)
CharTlistb:=function(G,CB)
```

```
local T,i;
T:=[];
for i in [1..Length(CB)] do
    T[i]:=ChiOf(CB[i][2],CB[i][1])/DegreeOfCharacter(CB[i][2]);
# ChiOf(chi,g) returns chi(g)
od;
return T;
end;

#-------------CharSb------------------------------
# Given group G and character basis CB, return
# S matrix of G's modular data
CharSb:=function(G,CB)
local i,j,k,l,x,y,S,a,b,g,h,chiA,chiB,val,temp,Coeff,D,
    row,p,KnC,Cg,Ka,Kb,setKb,Centralizers,X,Y,Blocks;

# Computing S using the permutation basis, then
# changing basis may be better for "large" groups
if Order(G)>32 then
    return CharSbyPerm(G,CB);
fi;
val:=0;
S:=[];
Coeff:=Inverse(Order(G));
for i in [1..Length(CB)] do
        S[i]:=[];
od;
D:=CharData(G,CB);
# D[i][1] = conjugacy class K(a) of CB[i][1]=a
# D[i][2] = list of x such that for each b in K(a) there is exactly one
# x with x^{-1}ax=b
for i in [1..Length(CB)] do
    chiA:=CB[i][2];
    a:=CB[i][1];
    Ka:=D[i][1];
    X:=D[i][2];
    Centralizers:=[];
    for k in [1..Length(Ka)] do
        Centralizers[k]:=Set(Centralizer(G,Ka[k]));
    od;
    for j in [i..Length(CB)] do
        val:=0;
        b:=CB[j][1];
        chiB:=CB[j][2];
        Kb:=D[j][1];
        Y:=D[j][2];
        for k in [1..Length(Ka)] do
            g:=Ka[k];
            x:=X[k];
            KnC:=Intersection(Kb,Centralizers[k]);
            for l in [1..Length(Kb)] do
                h:=Kb[l];
                if h in KnC then
                    y:=Y[l];
                    val:=val+ComplexConjugate(ChiOf(chiA,x*h*x^-1))
                        *ComplexConjugate(ChiOf(chiB,y*g*y^-1));
                fi;
            od;
        od;
        val:=val*Coeff;
        S[i][j]:=val;
        S[j][i]:=val;
    od;
# Memory saving
    Unbind(Centralizers);
    Unbind\[\](D,i);
od;
return S;
end;

#------------CharSbyPerm-----------------------
# Given group G and the character basis CB,
# compute the S matrix by computing S in the
# pemutation basis then changing basis
CharSbyPerm:=function(G,CB)
local PB,Sp,M;
PB:=PermBasis(G);
Sp:=PermSb(G,PB);
M:=COBMatrixPtoC(G,PB,CB);
return (M)*Sp*Inverse(M);
end;

#-------------COBMatrixPtoC----------------------
# Given G and the two bases, return the change-
# of-basis from perm basis (PB) to char basis (CB)
# See thesis for change-of-basis formula.
COBMatrixPtoC:=function(G,PB,CB)
local i,j,k,g,h,s,M,a,chi,Kg;
M:=0*IdentityMat(Length(PB));
for i in [1..Length(PB)] do
    g:=PB[i][1];
    h:=PB[i][2];
    Kg:=ConjugacyClass(G,g);
    s:=1/Order(Centralizer(Centralizer(G,g),h));
    for k in [1..Length(CB)] do
        # Only the (g,*) in CB have non-zero coefficients
        if IsConjugate(G,g,CB[k][1]) then
            chi:=CB[k][2];
            j:=FindIndexInCB(G,[CB[k][1],chi],CB);
            # FindIndexInCB retruns j such that CB[j]=[g,chi]

            # The prem and char bases need not use the same
            # set R of reps of conj classes of G
            if g<>CB[k][1] then
                a:=RepresentativeAction(G,g,CB[k][1]);
                M[j][i]:=s*ComplexConjugate(ChiOf(chi,Inverse(a)*h*a));
            else
                M[j][i]:=s*ComplexConjugate(ChiOf(chi,h));
            fi;
        fi;
    od;
od;
return M;
end;
```

A.2 Certificate algorithm

The following code is for Algorithms 4.3 and 4.4.

```
#------------------MatrixCertificateInitial-------------------------
# Given a matrix/edge-coloured graph M and an ordered partition P
# of its verticies, find the certificate with the restriction that
# P is respected.
MatrixCertificateInitial:=function(M,P)
local I,worst,n,result,i,Aut;
I:=MatrixElements(M);
P:=Refine(M,P,I,0);
n:=Length(M);
Aut:=[()];
result:=MatrixCertBack(M,P,I,[0,()],Aut);
return result[1];
end;


#--------------Refine-------------------------------
# Input:
#   M=matrix/edge-coloured graph
#   P=partition of nodes of M
#   I=sorted set of elements of M
#   A=starting partition blocks
# Refine the given partition P to an equitable
# partition.  If A=0, use P as the starting blocks.
# If A<>0, only use A.  A<>0 should only be used when
# we're in the "size 2 splitting" situation.
Refine:=function(M,P,I,A)
local B,S,T,n,U,Cp;
B:=ShallowCopy(P);
if A=0 then
    S:=Reversed(B);
else
    S:=Reversed(A);
fi;
U:=Flat(S);
while not IsEmpty(S) do
    n:=Length(S);
    T:=S[n];
    S:=S{[1..n-1]};
    # We need not consider a block if we've already considered
    # all the blocks that partition it.  When we use a block,
    # remove all its indicies from U.  If a block gets split,
    # the subblocks are candidates for splitters, so the indicies
    # are added to U.  Then as these subblocks get used as
    # splitters, their indicies are removed from U.  Once their
    # original block is dug down to in S, all of the indicies
    # will be gone since the subblocks have all been considered.
    if IsSubset(U,T) then
        U:=Difference(U,T);
        Cp:=ShallowCopy(B);
        SplitPartition(M,B,S,T,I,U);
        U:=Set(U);
    fi;
od;
return B;
end;

#---------------SplitPartition-------------------------
# Input:
#   M=matrix/edge-coloured graph
#   B=current ordered partition of indicies
#   S=set of blocks to be used for future splitting
#   T=block to be used to split B
#   I=sorted set of elements of M
#   U=set of 'unused' indicies
# Splits the partition B using the block T.  Adds
# any split blocks to S, and adds the indicies
# in any split blocks to U.
SplitPartition:=function(M,B,S,T,I,U)
local i,New;
i:=1;
while i<=Length(B) do
    New:=Split(M,B[i],T,I);
    if Length(New)>1 then
        Replace(New,B,i); # replaces B[i] with New
        i:=i+1;
        Append(S,Reversed(New));
        Append(U,Flat(New));
    fi;
    i:=i+1;
od;
end;

#------------Split-------------------------------------
# Given graph M, a set of indicies Block, a list of
# indicies T, and a (sorted) set of elements I of M.
# Splits Block according to T.  That is, order the elements
# of Block according to the vector of their coloured
# neighbours in T.  Group elements with the same
# coloured neighbours together in a subblock and
# order the subblocks by increasing size of neighbour vector.
# Return this ordered partition of Block.
Split:=function(M,Block,T,I)
local n,m,j,V,K,New,last,current,pos;
n:=Length(M);
m:=Length(I);
K:=[];
for j in Block do
```

```
    V:=-ColouredNeighbours(M,I,j,T);
    Add(V,j);
    Add(K,V);
od;
Sort(K, LexIgnoreLast);
# LexIgnoreLast compares lists lexicographically, but ignoring the last element
last:=K[1]{[1..m]};
New:=[[K[1][m+1]]];
pos:=1;
for j in [2..Length(K)] do
    current:=K[j]{[1..m]};
    last:=K[j-1]{[1..m]};
    if current<>last then
        pos:=pos+1;
        Add(New,[]);
    fi;
    Add(New[pos],K[j][m+1]);
od;
return New;
end;

#-----------ColouredNeighbours----------------------
# Given a matrix M, an ordered list L of its
# unique elements, an integer i, and a list of
# integers T, return a vector V where
#   V[i]=number of i's neighbours of type L[i],
#   (ie. number of times L[i] appears in M[i])i
#   only counting neighbours that are in locations
#   given by indicies in T
ColouredNeighbours:=function(M,L,i,T)
local j,k,V;
V:=0*[1..Length(L)];
for j in T do
    k:=Position(L,M[i][j]);
    if k=fail then continue; fi;
    V[k]:=V[k]+1;
od;
return V;
end;

#-----------------NumberMatrix------------------------
# Return the list formed by reading down the columns
# of the given matrix, left to right, stopping at
# the element on the diagonal.
NumberMatrix:=function(M)
local i,j,L;
L:=[];
for i in [1..Length(M)] do
    Append(L,M{[1..i]}{[i]});
od;
return Flat(L);
end;

#-----------------MatrixCertBack-------------------------
```

```
# Input:
#   M=matrix/edge-coloured graph
#   P=partition of M's indicies
#   I=sorted set of M's elements
#   Best[1]=best certificate found so far
#   Best[2]=perm that produces Best[1]
#   Aut=group of automorphisms of M found so far
# Backtracking algorithm for finding the certificate of
# a matrix/edge-coloured graph.  Uses the best certificate
# found so far and the automorphism group for pruning
# the search tree.
MatrixCertBack:=function(M,P,I,Best,Aut)
local i,k,j,c,C,n,Rest,N,part,Npart,l,Stab,pre,image,PR,prune;
# Find the first non-singleton in P
for i in [1..Length(P)] do
    if Length(P[i])>1 then
        break;
    fi;
od;
# If P is a total order...
if i=Length(M) then
    if Best[1]=0 then
        Best[1]:=NumberMatrix(M{Flat(P)}{Flat(P)});
        Best[2]:=(PermList(Flat(P)))^-1;
    fi;
    # If P defines the same cert as the best so far, then P
    # defines an automorphism of M, so we can remember it
    if NumberMatrix(M{Flat(P)}{Flat(P)})=Best[1] then
        Add(Aut,((PermList(Flat(P)))^-1)*(Best[2])^-1);
    fi;
    # P is a total order, so return the certificate
    return [NumberMatrix(M{Flat(P)}{Flat(P)}),(PermList(Flat(P)))^-1];
fi;
# The initial portion of P consisting of singletons defines
# a partial certificate
part:=Flat(P{[1..i-1]});
Npart:=NumberMatrix(M{part}{part});
n:=Length(Npart);
# Prune using the partial certificate
if Best[1]<>0 and Npart>Best[1]{[1..n]} then
    return Best;
fi;
# Run through the first non-singleton block, splitting
# off each element in turn and trying the partition
C:=P[i];
for j in [1..Length(C)] do
    c:=C[j];
    prune:=false;
    pre:=ShallowCopy(part);
    image:=ShallowCopy(part);
    image[i]:=c;
    for k in [1..j-1] do
        pre[i]:=C[k];
        # If we know of a automorphism of M that fixes the initial singleton
        # portion of P and replaces our current first-place element c with
        # a lower-index element of this block, we can prune since we've already
        # tried these lower-index elements in the first position
        if RepresentativeAction(Group(Aut),pre,image,OnTuples)<>fail then
            prune:=true;
            break;
        fi;
    od;
    # If we found the RepAction above, we can prune this branch so we continue
    # to the next candidate in C
    if prune then
        continue;
    fi;
    # Split the jth element off from C and place it first
    Rest:=Difference(C,[c]);
    Replace([[c],Rest],P,i);
    # Use size 2 block partitioning if applicable
    if Length(Rest)=1 then
        PR:=Refine(M,P,I,[[c],Rest]);
    else
        PR:=Refine(M,P,I,0);
    fi;
    # Compute the best cert for that partition
    N:=MatrixCertBack(M,PR,I,Best,Aut);
    if N[1]<Best[1] then
        Best:=ShallowCopy(N);
    fi;
    # Restore to the original partition
    Replace([C],P,i);
    Replace([],P,i+1);
od;
return Best;
end;
```

A.3 Code for showing weak equivalence

Below is the code for the backtracking algorithm used to find P such that $PS^GP^{-1} = S^H$ and $PT^GP^{-1} = T^H$. This is used to prove Theorem 4.5.

```
#---------PermSearch----------------
PermSearch:=function(Tg,Th,Sg,Sh)
local p,q,G,result;
G:=SymmetricGroup(Length(Tg));
p:=Random(G);
q:=Random(G);
result:=PermSearchBacktrack(Permuted(Tg,p),Permuted(Th,q),PermuteMatrix(Sg,p),
    PermuteMatrix(Sh,q),[],1,[1..Length(Tg)]);
if result[1]=false then
    return result;
else
    return [result[1],p*Inverse(PermList(result[2]))*Inverse(q)];
fi;
```

```
end;

----------PermSearchBacktrack------------
PermSearchBacktrack:=function(Tg,Th,Sg,Sh,perm,l,unchosen)
local i,result,copyunc;
if IsEmpty(unchosen) then
    if Sg{perm}{perm}=Sh and Tg{perm}=Th then
        return [ true, perm];
    else
        return [ false ];
    fi;
else
    for i in unchosen do
        perm[l]:=i;
        copyunc:=ShallowCopy(unchosen);
        RemoveSet(copyunc,i);
        if Sg{perm}{perm}=Sh{[1..l]}{[1..l]} and Tg{perm}=Th{[1..l]} then
            result:=PermSearchBacktrack(Tg,Th,Sg,Sh,perm,l+1,copyunc);
            if result[1]=true then
                return result;
            fi;
        fi;
    od;
fi;
Unbind(perm[l]);
return [false];
end;
```

A.4 Code for Proposition 5.1

```
#-----------------GroupDet-----------------
# Given G, calculate its group determinant
GroupDet:=function(G)
local I,LG,P,M,n,i,j,g,x;
n:=Order(G);
P:=PolynomialRing(Integers,n);
I:=IndeterminatesOfPolynomialRing(P);
LG:=0*[1..n];
i:=1;
for x in G do
    LG[i]:=x;
    i:=i+1;
od;
M:=0*IdentityMat(n);
for i in [1..n] do
    for j in [1..n] do
        g:=LG[i]*(LG[j])^(-1);
        M[i][j]:=I[Position(LG,g)];
    od;
od;
return Determinant(M);
end;
```

```
#---------CoefficientsOfMultivariatePolynomial------
# Given a multivariate polynomial P, return a list
# of its coefficients
CoefficientsOfMultivariatePolynomial:=function(P)
local i,M,N;
M:=CoeffOfMPRecurse(P,1);
N:=[];
for i in [1..Length(M)] do
    if IsZero(M[i])=false then
        Add(N,M[i]);
    fi;
od;
Sort(N);
return N;
end;

#----------CoeffOfMPRecurse-----------------------------
# Recursive function used in
# CoefficientsOfMultivariatePolynomial
CoeffOfMPRecurse:=function(L,n)
local i,M;
if (IsUnivariatePolynomial(L) and DegreeIndeterminate(L,n)>0) or
IsConstantRationalFunction(L) then
    return PolynomialCoefficientsOfPolynomial(L,n);
else
    M:=PolynomialCoefficientsOfPolynomial(L,n);
    for i in [1..Length(M)] do
        M[i]:=CoeffOfMPRecurse(ShallowCopy(M[i]),n+1);
    od;
    return Flat(M);
fi;
end;
```

A.5 Code for Proposition 5.10

```
#---------------CharEquivBases---------------------------
# Given groups g,h determine if their character bases are
# "character table equivalent".
CharEquivBases:=function(g,h)
local CTg,CTh,L,i,j;
CTg:=List(List(ConjugacyClasses(g),Representative),
    x->Irr(CharacterTable(Centralizer(g,x))));
CTh:=List(List(ConjugacyClasses(h),Representative),
    x->Irr(CharacterTable(Centralizer(h,x))));
if Length(CTg)<>Length(CTh) then
    return [false, "different sized bases"];
fi;
L:=0*[1..Length(CTg)];
for i in [1..Length(CTg)] do
    for j in [1..Length(CTh)] do
        if L[j]=1 then
            continue;
        fi;
        if TransformingPermutations(CTg[i],CTh[j])<>fail then
            L[j]:=1;
            break;
        fi;
    od;
od;
if Sum(L)=Length(CTg) then
    return [true];
else
    return [false, Sum(L), Length(CTg)];
fi;
end;
```

Bibliography

[Apo76] Tom M. Apostol. *Modular Functions and Dirichlet Series in Number Theory.* Number 41 in Graduate Texts in Mathematics. Springer-Verlag, 1976.

[Ban97] Peter Bantay. The Frobenius-Schur indicator in conformal field theory. *Phys. Lett. B*, 394(1-2):87–88, 1997.

[BBST01] Eiichi Bannai, Etsuko Bannai, Osamu Shimabukuro, and Makoto Tagami. Modular invariants of the modular data of finite groups (english). *Sūrikaisekikenkyūsho Kōkyūroku*, (1228):1–12, 2001. Codes, lattices, vertex operator algebras and finite groups (Japanese) (Kyoto, 2001).

[Bir75] Joan S. Birman. *Braids, links, and mapping class groups.* Princeton University Press, 1975.

[Bra63] Richard Brauer. Representations of finite groups. In T.L. Saaty, editor, *Lectures on Modern Mathematics*, pages 133–175. John Wiley & Sons, 1963.

[Cam99] Peter J. Cameron. *Permutation Groups.* Number 45 in London Mathematical Society Student Texts. Cambridge University Press, 1999.

[CGR00] Antoine Coste, Terry Gannon, and Phillippe Ruelle. Finite group modular data. *Nuclear Physics. B*, 581(3):679–717, 2000.

[Cun05] Michael Cuntz. *Fourier-Matrizen und Ringe mit Basis.* PhD thesis, Universität Kassel, 2005.

[Dad64] E.C. Dade. Answer to a question of R. Brauer. *Journal of Algebra*, 1:1–4, 1964.

[Dav00] A. A. Davydov. Finite groups with the same character tables, Drinfel'd algebras and Galois algebras. In *Algebra (Moscow, 1998)*, pages 99–111. de Gruyter, Berlin, 2000.

[DF99] David S. Dummit and Richard M. Foote. *Abstract Algebra.* John Wiley and Sons, second edition, 1999.

[Dor71] Larry Dornhoff. *Group Representation Theory.* Marcel Dekker, Inc., 1971.

[Eho93] Wolfgang Eholzer. Fusion algebras induced by representations of the modular group. *International Journal of Modern Physics A*, 8(20):3495–3507, 1993.

[Eho95] Wolfgang Eholzer. On the classification of modular fusion algebras. *Commun. Math. Phys.*, 172:623–659, 1995.

[FH91] William Fulton and Joe Harris. *Representation Theory.* Springer-Verlag, 1991.

[Fro68] Ferdinand Georg Frobenius. Über die Primfactoren der Gruppendeterminante. In J-P. Serre, editor, *Gesammelte Abhandlungen*, pages 38–77. Springer-Verlag, 1968.

[FS91] E. Formanek and D. Sibley. The group determinant determines the group. *Proc. Amer. Math. Soc.*, 112:649–656, 1991.

[Gan05] Terry Gannon. Modular data: the algebraic combinatorics of conformal field theory. *J. Algebraic Combin.*, 22(2):211–250, 2005.

[Gan06] Terry Gannon. *Moonshine beyond the Monster: The Bridge Connecting Algebra, Modular Forms and Physics.* Cambridge University Press, 2006.

[GAP05a] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.4*, 2005. (`http://www.gap-system.org`).

[GAP05b] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.4, package SmallGroups*, 2005. (`http://www.gap-system.org`).

[Gro97] Larry C. Grove. *Groups and Characters.* John Wiley & Sons, 1997.

[HJ95] H.-J. Hoehnke and K.W. Johnson. 3-characters are sufficient for the group determinant. *Contemporary Mathematics*, 184:193–206, 1995.

[JMS00] Kenneth W. Johnson, Sandro Mattarei, and Surinder Sehgal. Weak cayley tables. *J. London Math. Soc.*, 61(3):395–411, 2000.

[Joh91] K.W. Johnson. On the group determinant. *Math. Proc. Camb. Phil. Soc.*, 109:299–311, 1991.

[JS93] Kenneth W. Johnson and Surinder K. Sehgal. The 2-character table does not determine the group. *Proc. Amer. Math. Soc.*, 119(4):1021–1027, 1993.

[JS95] Kenneth W. Johnson and Surinder K. Sehgal. The 2-characters of a group and the group determinant. *Europ. J. Combinatorics*, 16:623–631, 1995.

[Kam02] Seiichi Kamada. *Braid and knot theory in dimension four*, volume 95 of *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI, 2002.

[KS99] Donald L. Kreher and Douglas R. Stinson. *Combinatorial Algorithms : Generation, Enumeration, and Search.* CRC Press, 1999.

[KSSB99] T.H. Koornwinder, B.J. Schroes, J.K. Slingerland, and F.A. Bais. Fourier transform and the Verlinde formula for the quantum double of a finite group. *J. Phys. A: Math. Gen.*, 32:8539–8549, 1999.

[Lan87] Serge Lang. *Elliptic Functions.* Springer-Verlag, 1987.

[Lus79] George Lusztig. Unipotent representations of a finite Chevalley group of type E_8. *Quart. J. Math. Oxford (2)*, 30:315–338, 1979.

[Man92] Richard Mansfield. A group determinant determines its group. *Proc. Amer. Math. Soc.*, 116(4):939–941, 1992.

[Mas95] Geoffrey Mason. The quantum double of a finite group and its role in conformal field theory. In *Groups '93 Galway/St. Andrews, Vol. 2*, number 212 in London Math. Soc. Lecture Note Ser., pages 405–417. Cambridge Univ. Press, 1995.

[NW76] Alexandre Nobs and Jürgen Wolfart. Die irreduziblen Darstellungen der Gruppen $SL_2(Z_p)$, insbesondere $SL_2(Z_p)$.II. *Comment. Math. Helv.*, 51(4):491–526, 1976.

[You93] J.W.A. Young. On the determination of groups whose order is a power of a prime. *American Journal of Mathematics*, 15:124–178, 1893.