Permutation Characters in GAP

Thomas Breuer

Lehrstuhl D für Mathematik
RWTH, 52056 Aachen, Germany

April 17th, 1999

This is a loose collection of examples of computations with permutation characters and possible permutation characters in GAP. We mainly use the GAP implementation of the algorithms to compute possible permutation characters that are described in [BP98], and information from the Atlas of Finite Groups [CCN+85].

A **possible permutation character** of a finite group G is a character satisfying the conditions listed in Section "Possible Permutation Characters" of the GAP Reference Manual.

1 Some Computations with M_{24}

We start with the sporadic simple Mathieu group $G = M_{24}$ in its natural action on 24 points.

```
gap> g:= MathieuGroup( 24 );;
gap> SetName( g, "m24" );
gap> Size( g );  IsSimple( g );  NrMovedPoints( g );
244823040
true
24
```

The permutation character `pi` of G corresponding to the action on the moved points is constructed. This action is 5-transitive.

```
gap> NrConjugacyClasses( g );
26
gap> pi:= NaturalCharacter( g );
Character( CharacterTable( m24 ), [ 24, 2, 1, 1, 4, 6, 0, 0, 3, 3, 0, 1, 1,
  8, 0, 2, 0, 0, 4, 2, 0, 0, 0, 1, 1, 0 ] )
gap> IsTransitive( pi );  Transitivity( pi );
true
5
gap> Display( pi );
m24
```

```
   2 10   .   .   .  2  3   .   .  1  1  3   1   1 10   2  3  7  9  7  4   2
   3  3   .   1   1  1  3   1   1  1  1  2   .   .  1   1  1  1  1  .  .   1
   5  1   .   1   1  1  1   .   .  .  .  .   .   .  .   .  .  .  1  .  .   .
   7  1   .   .   .  .  .   1   1  1  1  1   1   1  1   .  .  .  .  .  .   .
  11  1   1   .   .  .  .   .   .  .  .  .   .   .  .   .  .  .  .  .  .   .
  23  1   .   .   .  .  .   .   .  .  .  .   .   .  .   .  .  .  .  .  .   .

     1a 11a 15a 15b 5a 3a 21a 21b 7a 7b 3b 14a 14b 2a 12a 6a 4a 2b 4b 8a 12b

Y.1  24   2   1   1  4  6   .   .  3  3  .   1   1  8   .  2  .  .  4  2   .

   2  3  5   .   .   2
   3  1  1   .   .   .
   5  .  .   .   .   1
   7  .  .   .   .   .
  11  .  .   .   .   .
  23  .  .   1   1   .

     6b 4c 23a 23b 10a

Y.1   .  .   1   1   .
```

`pi` determines the permutation characters of the G-actions on related sets, for example `piop` on the set of ordered and `piup` on the set of unordered pairs of points.

```
gap> piop:= pi * pi;
Character( CharacterTable( m24 ), [ 576, 4, 1, 1, 16, 36, 0, 0, 9, 9, 0, 1,
  1, 64, 0, 4, 0, 0, 16, 4, 0, 0, 0, 1, 1, 0 ] )
gap> IsTransitive( piop );
false
gap> piup:= SymmetricParts( UnderlyingCharacterTable(pi), [ pi ], 2 )[1];
Character( CharacterTable( m24 ), [ 300, 3, 1, 1, 10, 21, 0, 0, 6, 6, 0, 2,
  2, 44, 1, 5, 4, 12, 12, 4, 0, 0, 0, 1, 1, 2 ] )
gap> IsTransitive( piup );
false
```

Clearly the action on unordered pairs is not transitive, since the pairs $[i, i]$ form an orbit of their own. There are exactly two G-orbits on the unordered pairs, hence the G-action on 2-sets of points is transitive.

```
gap> ScalarProduct( piup, TrivialCharacter( g ) );
2
gap> comb:= Combinations( [ 1 .. 24 ], 2 );;
gap> hom:= OperationHomomorphism( g, comb, OnSets );;
gap> pihom:= NaturalCharacter( hom );
Character( CharacterTable( m24 ), [ 276, 1, 0, 0, 6, 15, 0, 0, 3, 3, 0, 1, 1,
  36, 1, 3, 4, 12, 8, 2, 0, 0, 0, 0, 0, 2 ] )
gap> Transitivity( pihom );
1
```

In terms of characters, the permutation character `pihom` is the difference of `piup` and `pi` . Note that GAP does not know that this difference is in fact a character; in general this question is not easy to decide without knowing the irreducible characters of G, and up to now GAP has not computed the irreducibles.

```
gap> pi2s:= piup - pi;
VirtualCharacter( CharacterTable( m24 ), [ 276, 1, 0, 0, 6, 15, 0, 0, 3, 3,
  0, 1, 1, 36, 1, 3, 4, 12, 8, 2, 0, 0, 0, 0, 0, 2 ] )
gap> pi2s = pihom;
true
gap> HasIrr( g );  HasIrr( CharacterTable( g ) );
false
false
```

The point stabilizer in the action on 2-sets is in fact a maximal subgroup of G, which is isomorphic to the automorphism group $M_{22}:2$ of the Mathieu group M_{22}. Thus this permutation action is primitive. But we cannot apply `IsPrimitive` to the character `pihom` for getting this answer because primitivity of characters is defined in a different way, cf. `IsPrimitiveCharacter` in the GAP Reference Manual.

```
gap> IsPrimitive( g, comb, OnSets );
true
```

We could also have computed the transitive permutation character of degree 276 using the GAP library of character tables instead of the group G, since the character tables of G and all its maximal subgroups are available, together with the class fusions of the maximal subgroups into G.

```
gap> tbl:= CharacterTable( "M24" );
CharacterTable( "M24" )
gap> maxes:= Maxes( tbl );
[ "M23", "M22.2", "2^4:a8", "M12.2", "2^6:3.s6", "L3(4).3.2_2",
  "2^6:(psl(3,2)xs3)", "L2(23)", "L3(2)" ]
gap> s:= CharacterTable( maxes[2] );
CharacterTable( "M22.2" )
gap> TrivialCharacter( s )^tbl;
Character( CharacterTable( "M24" ), [ 276, 36, 12, 15, 0, 4, 8, 0, 6, 3, 0,
  3, 3, 2, 2, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0 ] )
```

Note that the sequence of conjugacy classes in the library table of G does in general not agree with the succession computed for the group.

2 All Possible Permutation Characters of M_{11}

We compute all possible permutation characters of the Mathieu group M_{11}, using the three different strategies available in GAP.

First we try the algorithm that enumerates all candidates via solving a system of inequalities, which is described in [BP98, Section 3.2].

```
gap> m11:= CharacterTable( "M11" );;
gap> SetName( m11, "m11" );
gap> perms:= PermChars( m11 );
[ Character( m11, [ 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 ] ),
  Character( m11, [ 11, 3, 2, 3, 1, 0, 1, 1, 0, 0 ] ),
  Character( m11, [ 12, 4, 3, 0, 2, 1, 0, 0, 1, 1 ] ),
  Character( m11, [ 22, 6, 4, 2, 2, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 55, 7, 1, 3, 0, 1, 1, 1, 0, 0 ] ),
  Character( m11, [ 66, 10, 3, 2, 1, 1, 0, 0, 0, 0 ] ),
  Character( m11, [ 110, 6, 2, 2, 0, 0, 2, 2, 0, 0 ] ),
  Character( m11, [ 110, 6, 2, 6, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 110, 14, 2, 2, 0, 2, 0, 0, 0, 0 ] ),
  Character( m11, [ 132, 12, 6, 0, 2, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 144, 0, 0, 0, 4, 0, 0, 0, 1, 1 ] ),
  Character( m11, [ 165, 13, 3, 1, 0, 1, 1, 1, 0, 0 ] ),
  Character( m11, [ 220, 4, 4, 0, 0, 4, 0, 0, 0, 0 ] ),
  Character( m11, [ 220, 12, 4, 4, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 220, 20, 4, 0, 0, 2, 0, 0, 0, 0 ] ),
  Character( m11, [ 330, 2, 6, 2, 0, 2, 0, 0, 0, 0 ] ),
  Character( m11, [ 330, 18, 6, 2, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 396, 12, 0, 4, 1, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 440, 8, 8, 0, 0, 2, 0, 0, 0, 0 ] ),
  Character( m11, [ 440, 24, 8, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 495, 15, 0, 3, 0, 0, 1, 1, 0, 0 ] ),
  Character( m11, [ 660, 4, 3, 4, 0, 1, 0, 0, 0, 0 ] ),
  Character( m11, [ 660, 12, 3, 0, 0, 3, 0, 0, 0, 0 ] ),
  Character( m11, [ 660, 12, 12, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 660, 28, 3, 0, 0, 1, 0, 0, 0, 0 ] ),
  Character( m11, [ 720, 0, 0, 0, 0, 0, 0, 0, 5, 5 ] ),
  Character( m11, [ 792, 24, 0, 0, 2, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 880, 0, 16, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 990, 6, 0, 2, 0, 0, 2, 2, 0, 0 ] ),
  Character( m11, [ 990, 6, 0, 6, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 990, 30, 0, 2, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 1320, 8, 6, 0, 0, 2, 0, 0, 0, 0 ] ),
  Character( m11, [ 1320, 24, 6, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 1584, 0, 0, 0, 4, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 1980, 12, 0, 4, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 1980, 36, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 2640, 0, 12, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 3960, 24, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( m11, [ 7920, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
gap> Length( perms );
39
```

Next we try the improved combinatorial approach that is sketched at the end of Section 3.2 in [BP98]. We get the same characters, except that they may be ordered in a different way; thus we compare the ordered lists.

```
gap> degrees:= DivisorsInt( Size( m11 ) );;
```

```
gap> perms2:= [];;
gap> for d in degrees do
>      Append( perms2, PermChars( m11, d ) );
>    od;
gap> Set( perms ) = Set( perms2 );
true
```

Finally, we try the algorithm that is based on Gaussian elimination and that is described in [BP98, Section 3.3].

```
gap> perms3:= [];;
gap> for d in degrees do
>      Append( perms3, PermChars( m11, rec( torso:= [ d ] ) ) );
>    od;
gap> Set( perms ) = Set( perms3 );
true
```

GAP provides two more functions to test properties of permutation characters. The first one yields no new information in our case, but the second excludes one possible permutation character; note that `TestPerm5` needs a p-modular Brauer table, and the GAP character table library contains all Brauer tables of M_{11}.

```
gap> newperms:= TestPerm4( m11, perms );;
gap> newperms = perms;
true
gap> newperms:= TestPerm5( m11, perms, m11 mod 11 );;
gap> newperms = perms;
false
gap> Difference( perms, newperms );
[ Character( m11, [ 220, 4, 4, 0, 0, 4, 0, 0, 0, 0 ] ) ]
```

GAP knows the table of marks of M_{11}, from which the permutation characters can be extracted. It turns out that M_{11} has 39 conjugacy classes of subgroups but only 36 different permutation characters, so three candidates computed above are in fact not permutation characters.

```
gap> tom:= TableOfMarks( "M11" );
TableOfMarks( "M11" )
gap> trueperms:= PermCharsTom( m11, tom );;
gap> Length( trueperms );  Length( Set( trueperms ) );
39
36
gap> Difference( perms, trueperms );
[ Character( m11, [ 220, 4, 4, 0, 0, 4, 0, 0, 0, 0 ] ),
  Character( m11, [ 660, 4, 3, 4, 0, 1, 0, 0, 0, 0 ] ),
  Character( m11, [ 660, 12, 3, 0, 0, 3, 0, 0, 0, 0 ] ) ]
```

3 The Action of $U_6(2)$ on the Cosets of M_{22}

We are interested in the permutation character of $U_6(2)$ (see [CCN+85, p. 115]) that corresponds to the action on the cosets of a M_{22} subgroup (see [CCN+85, p. 39]). The character tables of both the group and the point stabilizer are available in the GAP character table library, so we can compute class fusion and permutation character directly; note that if the class fusion is not stored on the table of the subgroup, in general one will not get a unique fusion but only a list of candidates for the fusion.

```
gap> u62:= CharacterTable( "U6(2)" );;
gap> m22:= CharacterTable( "M22" );;
gap> fus:= PossibleClassFusions( m22, u62 );
[ [ 1, 3, 7, 10, 14, 15, 22, 24, 24, 26, 33, 34 ],
  [ 1, 3, 7, 10, 14, 15, 22, 24, 24, 26, 34, 33 ],
  [ 1, 3, 7, 11, 14, 15, 22, 24, 24, 27, 33, 34 ],
  [ 1, 3, 7, 11, 14, 15, 22, 24, 24, 27, 34, 33 ],
  [ 1, 3, 7, 12, 14, 15, 22, 24, 24, 28, 33, 34 ],
  [ 1, 3, 7, 12, 14, 15, 22, 24, 24, 28, 34, 33 ] ]
gap> RepresentativesFusions( m22, fus, u62 );
[ [ 1, 3, 7, 10, 14, 15, 22, 24, 24, 26, 33, 34 ] ]
```

We see that there are six possible class fusions that are equivalent under table automorphisms of $U_6(2)$ and $M22$.

```
gap> cand:= Set( List( fus,
>  x -> Induced( m22, u62, [ TrivialCharacter( m22 ) ], x )[1] ) );
[ Character( CharacterTable( "U6(2)" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      0, 0, 48, 0, 16, 6, 0, 0, 0, 0, 0, 0, 6, 0, 2, 0, 0, 0, 4, 0, 0, 0, 0,
      1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( CharacterTable( "U6(2)" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      0, 48, 0, 0, 16, 6, 0, 0, 0, 0, 0, 0, 6, 0, 2, 0, 0, 4, 0, 0, 0, 0, 0,
      1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( CharacterTable( "U6(2)" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      48, 0, 0, 0, 16, 6, 0, 0, 0, 0, 0, 0, 6, 0, 2, 0, 4, 0, 0, 0, 0, 0, 0,
      1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
gap> PermCharInfo( u62, cand ).ATLAS;
[ "1a+22a+252a+616a+1155c+1386a+8064a+9240c",
  "1a+22a+252a+616a+1155b+1386a+8064a+9240b",
  "1a+22a+252a+616a+1155a+1386a+8064a+9240a" ]
gap> aut:= AutomorphismsOfTable( u62 );;  Size( aut );
24
gap> elms:= Filtered( Elements( aut ), x -> Order( x ) = 3 );
[ (10,11,12)(26,27,28)(40,41,42), (10,12,11)(26,28,27)(40,42,41) ]
gap> Position( cand, Permuted( cand[1], elms[1] ) );
3
gap> Position( cand, Permuted( cand[3], elms[1] ) );
2
```

The six fusions induce three different characters, they are conjugate under the action of the unique subgroup of order 3 in the group of table automorphisms of $U_6(2)$. The table automorphisms of order 3 are induced by group automorphisms of $U_6(2)$ (see [CCN$^+$85, p. 120]). As can be seen from the list of maximal subgroups of $U_6(2)$ in [CCN$^+$85, p. 115], the three induced characters are in fact permutation characters which belong to the three classes of maximal subgroups of type M_{22} in $U_6(2)$, which are permuted by an outer automorphism of order 3.

Now we want to compute the extension of the above permutation character to the group $U_6(2).2$, which corresponds to the action of this group on the cosets of a $M_{22}.2$ subgroup.

```
gap> u622:= CharacterTable( "U6(2).2" );;
gap> m222:= CharacterTable( "M22.2" );;
gap> fus:= PossibleClassFusions( m222, u622 );
[ [ 1, 3, 7, 10, 13, 14, 20, 22, 22, 24, 29, 38, 39, 42, 41, 46, 50, 53, 58,
      59, 59 ] ]
gap> cand:= Induced( m222, u622, [ TrivialCharacter( m222 ) ], fus[1] );
[ Character( CharacterTable( "U6(2).2" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      48, 0, 0, 16, 6, 0, 0, 0, 0, 0, 6, 0, 2, 0, 4, 0, 0, 0, 0, 1, 0, 0, 0,
      0, 0, 0, 0, 0, 1080, 72, 0, 48, 8, 0, 0, 0, 18, 0, 0, 0, 8, 0, 0, 2, 0,
      0, 0, 0, 2, 2, 0, 0, 0, 0, 0 ] ) ]
gap> PermCharInfo( u622, cand ).ATLAS;
[ "1a+22a+252a+616a+1155a+1386a+8064a+9240a" ]
```

We see that for the embedding of $M_{22}.2$ into $U_6(2).2$, the class fusion is unique, so we get a unique extension of one of the above permutation characters. This implies that exactly one class of maximal subgroups of type M_{22} extends to $M_{22}.2$ in a given group $U_6(2).2$.

4 Degree 20 736 Permutation Characters of $U_6(2)$

Now we show an alternative way to compute the characters dealt with in the previous example. This works also if the character table of the point stabilizer is not available. In this situation we can compute all those characters that have certain properties of permutation characters.

Of course this may take much longer than the above computations, which needed only a few seconds. (The following calculations may need several hours, depending on the computer used.)

```
gap> cand:= PermChars( u62, rec( torso := [ 20736 ] ) );
[ Character( CharacterTable( "U6(2)" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      0, 0, 48, 0, 16, 6, 0, 0, 0, 0, 0, 0, 6, 0, 2, 0, 0, 0, 4, 0, 0, 0, 0,
      1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( CharacterTable( "U6(2)" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      0, 48, 0, 0, 16, 6, 0, 0, 0, 0, 0, 0, 6, 0, 2, 0, 0, 4, 0, 0, 0, 0, 0,
      1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( CharacterTable( "U6(2)" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      48, 0, 0, 0, 16, 6, 0, 0, 0, 0, 0, 0, 6, 0, 2, 0, 4, 0, 0, 0, 0, 0, 0,
      1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
```

For the next step, that is, the computation of the extension of the permutation character to $U_6(2).2$, we may use the above information, since the values on the inner classes are prescribed.

The question which of the three candidates for $U_6(2)$ extends to $U_6(2).2$ depends on the choice of the class fusion of $U_6(2)$ into $U_6(2).2$. With respect to the class fusion that is stored on the GAP library table, the third candidate extends, as can be seen from the fact that this one is invariant under the permutation of conjugacy classes of $U_6(2)$ that is induced by the action of the chosen supergroup $U_6(2).2$.

```
gap> u622:= CharacterTable( "U6(2).2" );;
gap> inv:= InverseMap( GetFusionMap( u62, u622 ) );
[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, [ 11, 12 ], 13, 14, 15, [ 16, 17 ], 18, 19,
  20, 21, 22, 23, 24, 25, 26, [ 27, 28 ], [ 29, 30 ], 31, 32, [ 33, 34 ],
  [ 35, 36 ], 37, [ 38, 39 ], 40, [ 41, 42 ], 43, 44, [ 45, 46 ] ]
gap> ext:= List( cand, x -> CompositionMaps( x, inv ) );
[ [ 20736, 0, 384, 0, 0, 0, 54, 0, 0, 0, [ 0, 48 ], 0, 16, 6, 0, 0, 0, 0, 0,
      6, 0, 2, 0, 0, [ 0, 4 ], 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 ],
  [ 20736, 0, 384, 0, 0, 0, 54, 0, 0, 0, [ 0, 48 ], 0, 16, 6, 0, 0, 0, 0, 0,
      6, 0, 2, 0, 0, [ 0, 4 ], 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 ],
  [ 20736, 0, 384, 0, 0, 0, 54, 0, 0, 48, 0, 0, 16, 6, 0, 0, 0, 0, 0, 6, 0,
      2, 0, 4, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 ] ]
gap> cand:= PermChars( u622, rec( torso:= ext[3] ) );
[ Character( CharacterTable( "U6(2).2" ), [ 20736, 0, 384, 0, 0, 0, 54, 0, 0,
      48, 0, 0, 16, 6, 0, 0, 0, 0, 0, 6, 0, 2, 0, 4, 0, 0, 0, 0, 1, 0, 0, 0,
      0, 0, 0, 0, 0, 1080, 72, 0, 48, 8, 0, 0, 0, 18, 0, 0, 0, 8, 0, 0, 2, 0,
      0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0 ] ) ]
```

5 Degree 57 572 775 Permutation Characters of $O_8^+(3)$

The group $O_8^+(3)$ (see [CCN+85, p. 140]) contains a subgroup of type $2^{3+6}.L_3(2)$, which extends to a maximal subgroup U in $O_8^+(3).3$. For the computation of the permutation character, we cannot use explicit induction since the table of U is not available in the GAP table library.

Since $U \cap O_8^+(3)$ is contained in a $O_8^+(2)$ subgroup of $O_8^+(3)$, we can try to find the permutation character of $O_8^+(2)$ corresponding to the action on the cosets of $U \cap O_8^+(3)$, and then induce this character to $O_8^+(3)$.

This kind of computations becomes more difficult with increasing degree, so we try to reduce the problem further. In fact, the $2^{3+6}.L_3(2)$ group is contained in a $2^6 : A_8$ subgroup of $O_8^+(2)$, in which the index is only 15; the unique possible permutation character of this degree can be read off immediately.

Induction to $O_8^+(3)$ through the chain of subgroups is possible provided the class fusions are available. There are 24 possible fusions from $O_8^+(2)$ into $O_8^+(3)$, which are all equivalent w.r.t. table automorphisms of $O_8^+(3)$. If we later want to consider the extension of the permutation character in question to $O_8^+(3).3$ then we have to choose a fusion of an $O_8^+(2)$ subgroup that does **not** extend to $O_8^+(2).3$. But if for example our question is just whether the resulting permutation character is multiplicity-free then this can be decided already from the permutation character of $O_8^+(3)$.

```
gap> o8p3:= CharacterTable("O8+(3)");;
gap> Size( o8p3 ) / (2^9*168);
57572775
gap> o8p2:= CharacterTable( "O8+(2)" );;
gap> fus:= PossibleClassFusions( o8p2, o8p3 );;
gap> Length( fus );
24
gap> rep:= RepresentativesFusions( o8p2, fus, o8p3 );
[ [ 1, 5, 2, 3, 4, 5, 7, 8, 12, 16, 17, 19, 23, 20, 21, 22, 23, 24, 25, 26,
      37, 38, 42, 31, 32, 36, 49, 52, 51, 50, 43, 44, 45, 53, 55, 56, 57, 71,
      71, 71, 72, 73, 74, 78, 79, 83, 88, 89, 90, 94, 100, 101, 105 ] ]
gap> fus:= rep[1];;
```

```
gap> Size( o8p2 ) / (2^9*168);
2025
gap> sub:= CharacterTable( "2^6:A8" );;
gap> subfus:= GetFusionMap( sub, o8p2 );
[ 1, 3, 2, 2, 4, 5, 6, 13, 3, 6, 12, 13, 14, 7, 21, 24, 11, 30, 29, 31, 13,
  17, 15, 16, 14, 17, 36, 37, 18, 41, 24, 44, 48, 28, 33, 32, 34, 35, 35, 51,
  51 ]
gap> fus:= CompositionMaps( fus, subfus );
[ 1, 2, 5, 5, 3, 4, 5, 23, 2, 5, 19, 23, 20, 7, 37, 31, 17, 50, 51, 43, 23,
  23, 21, 22, 20, 23, 56, 57, 24, 72, 31, 78, 89, 52, 45, 44, 53, 55, 55,
  100, 100 ]
gap> Size( sub ) / (2^9*168);
15
gap> List( Irr( sub ), Degree );
[ 1, 7, 14, 20, 21, 21, 21, 28, 35, 45, 45, 56, 64, 70, 28, 28, 35, 35, 35,
  35, 70, 70, 70, 70, 140, 140, 140, 140, 140, 210, 210, 252, 252, 280, 280,
  315, 315, 315, 315, 420, 448 ]
gap> cand:= PermChars( sub, 15 );
[ Character( CharacterTable( "2^6:A8" ), [ 15, 15, 15, 7, 7, 7, 7, 7, 3, 3,
      3, 3, 3, 0, 0, 0, 3, 3, 3, 3, 3, 3, 3, 3, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1,
      1, 1, 1, 1, 1, 0, 0 ] ) ]
gap> ind:= Induced( sub, o8p3, cand, fus );
[ Character( CharacterTable( "O8+(3)" ), [ 57572775, 59535, 59535, 59535,
      3591, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2187, 0, 27, 135, 135, 135, 243,
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 27, 27, 27, 0,
      0, 0, 0, 27, 27, 27, 27, 0, 8, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
gap> o8p33:= CharacterTable( "O8+(3).3" );;
gap> inv:= InverseMap( GetFusionMap( o8p3, o8p33 ) );
[ 1, [ 2, 3, 4 ], 5, 6, [ 7, 8, 9 ], [ 10, 11, 12 ], 13, [ 14, 15, 16 ], 17,
  18, 19, [ 20, 21, 22 ], 23, [ 24, 25, 26 ], [ 27, 28, 29 ], 30,
  [ 31, 32, 33 ], [ 34, 35, 36 ], [ 37, 38, 39 ], [ 40, 41, 42 ],
  [ 43, 44, 45 ], 46, [ 47, 48, 49 ], 50, [ 51, 52, 53 ], 54, 55, 56, 57,
  [ 58, 59, 60 ], [ 61, 62, 63 ], 64, [ 65, 66, 67 ], 68, [ 69, 70, 71 ],
  [ 72, 73, 74 ], [ 75, 76, 77 ], [ 78, 79, 80 ], [ 81, 82, 83 ], 84, 85,
  [ 86, 87, 88 ], [ 89, 90, 91 ], [ 92, 93, 94 ], 95, 96, [ 97, 98, 99 ],
  [ 100, 101, 102 ], [ 103, 104, 105 ], [ 106, 107, 108 ], [ 109, 110, 111 ],
  [ 112, 113, 114 ] ]
gap> ext:= CompositionMaps( ind[1], inv );
[ 57572775, 59535, 3591, 0, 0, 0, 0, 0, 2187, 0, 27, 135, 243, 0, 0, 0, 0, 0,
  0, 0, 27, 0, 0, 27, 27, 0, 8, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ]
gap> perms:= PermChars( o8p33, rec( torso:= ext ) );
[ Character( CharacterTable( "O8+(3).3" ), [ 57572775, 59535, 3591, 0, 0, 0,
      0, 0, 2187, 0, 27, 135, 243, 0, 0, 0, 0, 0, 0, 0, 27, 0, 0, 27, 27, 0,
      8, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
      0, 0, 3159, 3159, 243, 243, 39, 39, 3, 3, 0, 0, 0, 0, 0, 0, 0, 0, 3, 3,
      3, 3, 3, 3, 0, 0, 0, 0, 0, 0, 2, 2, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
     ] ) ]
```

```
gap> PermCharInfo( o8p33, perms ).ATLAS;
[ "1a+780aabb+2457a+2808abc+9450aaabbcc+18200abcdddef+24192a+54600a^{5}b+70200\
aabb+87360ab+139776a^{5}+147420a^{4}b^{4}+163800ab+184275aabc+199017aa+218700a\
+245700a+291200aef+332800a^{4}b^{5}c^{5}+491400aaabcd+531441a^{5}b^{4}c^{4}+55\
2825a^{4}+568620aabb+698880a^{4}b^{4}+716800aaabbccdddeeff+786240aabb+873600aa\
+998400aa+1257984a^{6}+1397760aa" ]
```

6 The Action of $O_7(3).2$ on the Cosets of $2^7.S_7$

We want to know whether the permutation character of $O_7(3).2$ (see [CCN+85, p. 108]) on the cosets of its maximal subgroup U of type $2^7.S_7$ is multiplicity-free.

As in the previous examples, first we try to compute the permutation character of the simple group $O_7(3)$. It turns out that the direct computation of all candidates from the degree is very time consuming. But we can use for example the additional information provided by the fact that U contains an A_7 subgroup. We compute the possible class fusions.

```
gap> o73:= CharacterTable( "O7(3)" );;
gap> a7:= CharacterTable( "A7" );;
gap> fus:= PossibleClassFusions( a7, o73 );
[ [ 1, 3, 6, 10, 15, 16, 24, 33, 33 ], [ 1, 3, 7, 10, 15, 16, 22, 33, 33 ] ]
```

We cannot decide easily which fusion is the right one, but already the fact that no other fusions are possible gives us some information about impossible constituents of the permutation character we want to compute.

```
gap> ind:= List( fus,
>      x -> Induced( a7, o73, [ TrivialCharacter( a7 ) ], x )[1] );;
gap> mat:= MatScalarProducts( o73, Irr( o73 ), ind );;
gap> sum:= Sum( mat );
[ 2, 6, 2, 0, 8, 6, 2, 4, 4, 8, 3, 0, 4, 4, 9, 3, 5, 0, 0, 9, 0, 10, 5, 6,
  15, 1, 12, 1, 15, 7, 2, 4, 14, 16, 0, 12, 12, 7, 8, 8, 14, 12, 12, 14, 6,
  6, 20, 16, 12, 12, 12, 10, 10, 12, 12, 8, 12, 6 ]
gap> const:= Filtered( [ 1 .. Length( sum ) ], x -> sum[x] <> 0 );
[ 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17, 20, 22, 23, 24, 25, 26,
  27, 28, 29, 30, 31, 32, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46,
  47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58 ]
gap> Length( const );
52
gap> const:= Irr( o73 ){ const };;
gap> rat:= RationalizedMat( const );;
```

But much more can be deduced from the fact that certain zeros of the permutation character can be predicted.

```
gap> names:= ClassNames( o73 );
[ "1a", "2a", "2b", "2c", "3a", "3b", "3c", "3d", "3e", "3f", "3g", "4a",
  "4b", "4c", "4d", "5a", "6a", "6b", "6c", "6d", "6e", "6f", "6g", "6h",
```

```
  "6i", "6j", "6k", "6l", "6m", "6n", "6o", "6p", "7a", "8a", "8b", "9a",
  "9b", "9c", "9d", "10a", "10b", "12a", "12b", "12c", "12d", "12e", "12f",
  "12g", "12h", "13a", "13b", "14a", "15a", "18a", "18b", "18c", "18d", "20a"
 ]
gap> List( fus, x -> names{ x } );
[ [ "1a", "2b", "3b", "3f", "4d", "5a", "6h", "7a", "7a" ],
  [ "1a", "2b", "3c", "3f", "4d", "5a", "6f", "7a", "7a" ] ]
gap> torso:= [ 28431 ];;
gap> zeros:= [ 5, 8, 9, 11, 17, 20, 23, 28, 29, 32, 36, 37, 38,
>              43, 46, 47, 48, 53, 54, 55, 56, 57, 58 ];;
gap> for i in zeros do
>      Print( names[i], " " );
>    od; Print( "\n" );
3a 3d 3e 3g 6a 6d 6g 6l 6m 6p 9a 9b 9c 12b 12e 12f 12g 15a 18a 18b 18c 18d 20a\
```

Every order 3 element of U lies in an A_7 subgroup of U, so among the classes of element order 3, at most the classes 3B, 3C, and 3F can have nonzero permutation character values. The excluded classes of element order 6 are the square roots of the excluded order 3 elements, likewise the given classes of element orders 9, 12, and 18 are excluded. The character value on 20A must be zero because U does not contain elements of this order. So we enter the additional information about these zeros.

```
gap> for i in zeros do
>      torso[i]:= 0;
>    od;
gap> torso;
[ 28431,,,, 0,,, 0, 0,, 0,,,,,, 0,,, 0,,, 0,,,,, 0, 0,,, 0,,,, 0, 0, 0,,,,, 0,
  ,, 0, 0, 0,,,,, 0, 0, 0, 0, 0, 0 ]
gap> perms:= PermChars( o73, rec( torso:= torso, chars:= rat ) );
[ Character( CharacterTable( "O7(3)" ), [ 28431, 567, 567, 111, 0, 0, 243, 0,
      0, 81, 0, 15, 3, 27, 15, 6, 0, 0, 27, 0, 3, 27, 0, 0, 0, 3, 9, 0, 0, 3,
      3, 0, 4, 1, 1, 0, 0, 0, 0, 2, 2, 3, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
      0, 0, 0, 0 ] ) ]
gap> PermCharInfo( o73, perms ).ATLAS;
[ "1a+78a+168a+182a+260ab+1092a+2457a+2730a+4095b+5460a+11648a" ]
```

We see that this character is already multiplicity free, so this holds also for its extension to $O_7(3).2$, and we need not compute this extension. (Of course we could compute it in the same way as in the examples above.)

7 The Action of $O_8^+(3).2_1$ on the Cosets of $2^7.A_8$

We are interested in the permutation character of $O_8^+(3).2_1$ that corresponds to the action on the cosets of a subgroup of type $2^7.A_8$. The intersection of the point stabilizer with the simple group $O_8^+(3)$ is of type $2^6.A_8$. First we compute the class fusion of these groups, modulo problems with ambiguities due to table automorphisms.

```
gap> o8p3:= CharacterTable( "O8+(3)" );;
gap> o8p2:= CharacterTable( "O8+(2)" );;
```

```
gap> fus:= PossibleClassFusions( o8p2, o8p3 );;
gap> NamesOfFusionSources( o8p2 );
[ "2^8:O8+(2)", "2^6:A8", "2.O8+(2)", "S6(2)" ]
gap> sub:= CharacterTable( "2^6:A8" );;
gap> subfus:= GetFusionMap( sub, o8p2 );
[ 1, 3, 2, 2, 4, 5, 6, 13, 3, 6, 12, 13, 14, 7, 21, 24, 11, 30, 29, 31, 13,
  17, 15, 16, 14, 17, 36, 37, 18, 41, 24, 44, 48, 28, 33, 32, 34, 35, 35, 51,
  51 ]
gap> fus:= List( fus, x -> CompositionMaps( x, subfus ) );;
gap> fus:= Set( fus );;
gap> Length( fus );
24
```

The ambiguities due to Galois automorphisms disappear when we are looking for the permutation characters induced by the fusions.

```
gap> ind:= List( fus, x -> Induced( sub, o8p3,
>                              [ TrivialCharacter( sub ) ], x )[1] );;
gap> ind:= Set( ind );;
gap> Length( ind );
6
```

Now we try to extend the candidates to $O_8^+(3).2_1$; the choice of the fusion of $O_8^+(3)$ into $O_8^+(3).2_1$ determines which of the candidates may extend.

```
gap> o8p32:= CharacterTable( "O8+(3).2_1" );;
gap> fus:= GetFusionMap( o8p3, o8p32 );;
gap> ext:= List( ind, x -> CompositionMaps( x, InverseMap( fus ) ) );;
gap> ext:= Filtered( ext, x -> ForAll( x, IsInt ) );
[ [ 3838185, 17577, 8505, 8505, 873, 0, 0, 0, 0, 6561, 0, 0, 729, 0, 9, 105,
      45, 45, 105, 30, 0, 0, 0, 0, 0, 0, 0, 0, 0, 189, 0, 0, 0, 9, 9, 27, 27,
      0, 0, 27, 9, 0, 8, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0,
      0, 0, 9, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 0,
      0, 0 ], [ 3838185, 17577, 8505, 8505, 873, 0, 6561, 0, 0, 0, 0, 0, 729,
      0, 9, 105, 45, 45, 105, 30, 0, 0, 0, 0, 0, 0, 189, 0, 0, 0, 9, 0, 0, 0,
      9, 27, 27, 0, 0, 9, 27, 0, 8, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0,
      0, 0, 9, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 0,
      0, 0, 0, 0, 0 ] ]
```

We compute the extensions of the first candidate; the other belongs to another class of subgroups, which is the image under an outer automorphism. (These calculations may need about one hour, depending on the computer used.)

```
gap> perms:= PermChars( o8p32, rec( torso:= ext[1] ) );
[ Character( CharacterTable( "O8+(3).2_1" ),
    [ 3838185, 17577, 8505, 8505, 873, 0, 0, 0, 0, 6561, 0, 0, 729, 0, 9,
      105, 45, 45, 105, 30, 0, 0, 0, 0, 0, 0, 0, 0, 0, 189, 0, 0, 0, 9, 9,
      27, 27, 0, 0, 27, 9, 0, 8, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0,
      0, 0, 0, 0, 9, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 6, 0, 0, 0,
```

```
        0, 0, 0, 0, 3159, 1575, 567, 63, 87, 15, 0, 0, 45, 0, 81, 9, 27, 0, 0,
        3, 3, 3, 3, 5, 5, 0, 0, 0, 4, 0, 0, 27, 0, 9, 0, 0, 15, 0, 3, 0, 0, 2,
        0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
gap> PermCharInfo( o8p32, perms ).ATLAS;
[ "1a+260abc+520ab+819a+2808b+9450aab+18200a+23400ac+29120b+36400aab+46592abce\
+49140d+66339a+98280ab+163800a+189540d+232960d+332800ab+368550a+419328a+531441\
ab" ]
```

Now we repeat the calculations for $O_8^+(3).2_2$ instead of $O_8^+(3).2_1$.

```
gap> o8p32:= CharacterTable( "O8+(3).2_2" );;
gap> fus:= GetFusionMap( o8p3, o8p32 );;
gap> ext:= List( ind, x -> CompositionMaps( x, InverseMap( fus ) ) );;
gap> ext:= Filtered( ext, x -> ForAll( x, IsInt ) );;
gap> perms:= PermChars( o8p32, rec( torso:= ext[1] ) );
[ Character( CharacterTable( "O8+(3).2_2" ), [ 3838185, 17577, 8505, 873, 0,
        0, 0, 6561, 0, 0, 0, 0, 729, 0, 9, 105, 45, 105, 30, 0, 0, 0, 0, 0, 0,
        189, 0, 0, 0, 9, 0, 9, 27, 0, 0, 0, 27, 27, 9, 0, 8, 1, 1, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 9, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0,
        0, 0, 0, 0, 6, 0, 0, 0, 0, 0, 0, 0, 199017, 2025, 297, 441, 73, 9, 0,
        1215, 0, 0, 0, 0, 0, 81, 0, 0, 0, 0, 27, 27, 0, 1, 9, 12, 0, 0, 45, 0,
        0, 1, 0, 0, 3, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 1, 0, 0, 0, 0, 0, 0
      ] ) ]
gap> PermCharInfo( o8p32, perms ).ATLAS;
[ "1a+260aac+520ab+819a+2808a+9450aaa+18200accee+23400ac+29120a+36400a+46592aa\
+49140c+66339a+93184a+98280ab+163800a+184275ac+189540c+232960c+332800aa+419328\
a+531441aa" ]
```

8 The Action of $S_4(4).4$ on the Cosets of $5^2.[2^5]$

We want to know whether the permutation character corresponding to the action of $S_4(4).4$ (see [CCN+85, p. 44]) on the cosets of its maximal subgroup of type $5^2 : [2^5]$ is multiplicity free.

The library names of subgroups for which the class fusions are stored are listed as value of the attribute `NamesOfFusionSources`, and for groups whose isomorphism type is not determined by the name this is the recommended way to find out whether the table of the subgroup is contained in the GAP library and known to belong to this group. (It might be that a table with such a name is contained in the library but belongs to another group, and it may also be that the table of the group is contained in the library –with any name– but it is not known that this group is isomorphic to a subgroup of $S_4(4).4$.

```
gap> s444:= CharacterTable( "S4(4).4" );;
gap> NamesOfFusionSources( s444 );
[ "S4(4)", "S4(4).2" ]
```

So we cannot simply fetch the table of the subgroup. As in the previous examples, we compute the possible permutation characters.

```
gap> perms:= PermChars( s444, rec( torso:= [ Size( s444 ) / ( 5^2*2^5 ) ] ) );
```

```
[ Character( CharacterTable( "S4(4).4" ), [ 4896, 384, 96, 0, 16, 32, 36, 16,
      0, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ),
  Character( CharacterTable( "S4(4).4" ), [ 4896, 192, 32, 0, 0, 8, 6, 1, 0,
      2, 0, 0, 36, 0, 12, 0, 0, 0, 1, 0, 6, 6, 2, 2, 0, 0, 0, 0, 1, 1 ] ),
  Character( CharacterTable( "S4(4).4" ), [ 4896, 240, 64, 0, 8, 8, 36, 16,
      0, 0, 0, 0, 0, 12, 8, 0, 4, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
```

So there are three candidates. None of them is multiplicity free, so we need not decide which of the candidates actually belongs to the group $5^2 : [2^5]$ we have in mind.

```
gap> PermCharInfo( s444, perms ).ATLAS;
[ "1abcd+50abcd+153abcd+170a^{4}b^{4}+680aabb",
  "1a+50ac+153a+170aab+256a+680abb+816a+1020a",
  "1ac+50ac+68a+153abcd+170aabbb+204a+680abb+1020a" ]
```

(If we would be interested which candidate is the right one, we could for example look at the intersection with $S_4(4)$, and hope for a contradiction to the fact that the group must lie in a $(A_5 \times A_5) : 2$ subgroup.)

9 The Action of Co_1 on the Cosets of Involution Centralizers

We compute the permutation characters of the sporadic simple Conway group Co_1 (see [CCN+85, p. 180]) corresponding to the actions on the cosets of involution centralizers. Equivalently, we are interested in the action of Co_1 on conjugacy classes of involutions. These characters can be computed as follows. First we take the table of Co_1.

```
gap> t:= CharacterTable( "Co1" );
CharacterTable( "Co1" )
```

The centralizer of each `2A` element is a maximal subgroup of Co_1. This group is also contained in the table library. So we can compute the permutation character by explicit induction, and the decomposition in irreducibles is computed with the command `PermCharInfo`.

```
gap> s:= CharacterTable( Maxes( t )[5] );
CharacterTable( "2^(1+8)+.O8+(2)" )
gap> ind:= Induced( s, t, [ TrivialCharacter( s ) ] );;
gap> PermCharInfo( t, ind ).ATLAS;
[ "1a+299a+17250a+27300a+80730a+313950a+644644a+2816856a+5494125a+12432420a+24\
794000a" ]
```

The centralizer of a `2B` element is not maximal. First we compute which maximal subgroup can contain it. The character tables of all maximal subgroups of Co_1 are contained in the GAP's table library, so we may take these tables and look at the group orders.

```
gap> centorder:= SizesCentralizers( t )[3];;
gap> maxes:= List( Maxes( t ), CharacterTable );;
gap> cand:= Filtered( maxes, x -> Size( x ) mod centorder = 0 );
[ CharacterTable( "(A4xG2(4)):2" ) ]
gap> u:= cand[1];;
gap> index:= Size( u ) / centorder;
3
```

So there is a unique class of maximal subgroups containing the centralizer of a 2B element, as a subgroup of index 3. We compute the unique permutation character of degree 3 of this group, and induce this character to G.

```
gap> subperm:= PermChars( u, rec( degree := index, bounds := false ) );
[ Character( CharacterTable( "(A4xG2(4)):2" ),
    [ 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
      1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 3, 3, 3, 3, 3,
      3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 0, 0, 0, 0, 0, 0, 0, 0,
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
      1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 ] ) ]
gap> subperm = PermChars( u, rec( torso := [ 3 ] ) );
true
gap> ind:= Induced( u, t, subperm );
[ Character( CharacterTable( "Co1" ), [ 2065694400, 181440, 119408, 38016,
      2779920, 0, 0, 378, 30240, 864, 0, 720, 316, 80, 2520, 30, 0, 6480,
      1508, 0, 0, 0, 0, 0, 38, 18, 105, 0, 600, 120, 56, 24, 0, 12, 0, 0, 0,
      120, 48, 18, 0, 0, 6, 0, 360, 144, 108, 0, 0, 10, 0, 0, 0, 0, 0, 4, 2,
      3, 9, 0, 0, 15, 3, 0, 0, 4, 4, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 12,
      8, 0, 6, 0, 0, 3, 0, 1, 0, 3, 3, 0, 0, 0, 0, 0, 0, 0, 3, 0 ] ) ]
gap> PermCharInfo( t, ind ).ATLAS;
[ "1a+1771a+8855a+27300aa+313950a+345345a+644644aa+871884aaa+1771000a+2055625a\
+4100096a+7628985a+9669660a+12432420aa+21528000aa+23244375a+24174150aa+2479400\
0a+31574400aa+40370176a+60435375a+85250880aa+100725625a+106142400a+150732800a+\
184184000a+185912496a+207491625a+299710125a+302176875a" ]
```

Finally, we try the same for the centralizer of a 2C element.

```
gap> centorder:= SizesCentralizers( t )[4];;
gap> cand:= Filtered( maxes, x -> Size( x ) mod centorder = 0 );
[ CharacterTable( "Co2" ), CharacterTable( "2^11:M24" ) ]
```

The group order excludes all except two classes of maximal subgroups. But the 2C centralizer cannot lie in Co_2 because the involution centralizers in Co_2 are too small.

```
gap> u:= cand[1];;
gap> GetFusionMap( u, t );
[ 1, 2, 2, 4, 7, 6, 9, 11, 11, 10, 11, 12, 14, 17, 16, 21, 23, 20, 22, 22,
  24, 28, 30, 33, 31, 32, 33, 33, 37, 42, 41, 43, 44, 48, 52, 49, 53, 55, 53,
  52, 54, 60, 60, 60, 64, 65, 65, 67, 66, 70, 73, 72, 78, 79, 84, 85, 87, 92,
  93, 93 ]
gap> centorder;
389283840
gap> SizesCentralizers( u )[4];
1474560
```

So we try the second candidate.

```
gap> u:= cand[2];
CharacterTable( "2^11:M24" )
gap> index:= Size( u ) / centorder;
1288
gap> subperm:= PermChars( u, rec( torso := [ index ] ) );
[ Character( CharacterTable( "2^11:M24" ), [ 1288, 1288, 1288, 56, 56, 56,
      56, 56, 56, 48, 48, 48, 48, 48, 10, 10, 10, 10, 7, 7, 8, 8, 8, 8, 8, 8,
      4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 3, 3, 3, 2, 2, 2, 2, 2, 2, 3, 3, 3, 0,
      0, 0, 0, 2, 2, 2, 2, 3, 3, 3, 1, 1, 2, 2, 2, 2, 1, 1, 0, 0, 0, 0, 0, 0,
      0, 0, 0, 0, 0, 0 ] ) ]
gap> subperm = PermChars( u, rec( degree:= index, bounds := false ) );
true
gap> ind:= Induced( u, t, subperm );
[ Character( CharacterTable( "Co1" ), [ 10680579000, 1988280, 196560, 94744,
      0, 17010, 0, 945, 7560, 3432, 2280, 1728, 252, 308, 0, 225, 0, 0, 0,
      270, 0, 306, 0, 46, 45, 25, 0, 0, 120, 32, 12, 52, 36, 36, 0, 0, 0, 0,
      0, 45, 15, 0, 9, 3, 0, 0, 0, 0, 18, 0, 30, 0, 6, 18, 0, 3, 5, 0, 0, 0,
      0, 0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 3, 0, 0, 0, 0, 1, 0, 0, 0, 0, 6, 0, 2,
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ] ) ]
gap> PermCharInfo( t, last ).ATLAS;
[ "1a+17250aa+27300a+80730aa+644644aaa+871884a+1821600a+2055625aaa+2816856a+54\
94125a^{4}+12432420aa+16347825aa+23244375a+24174150aa+24667500aa+24794000aaa+3\
1574400a+40370176a+55255200a+66602250a^{4}+83720000aa+85250880aaa+91547820aa+1\
06142400a+150732800a+184184000aaa+185912496aaa+185955000aaa+207491625aaa+21554\
7904aa+241741500aaa+247235625a+257857600aa+259008750a+280280000a+302176875a+32\
6956500a+387317700a+402902500a+464257024a+469945476b+502078500a+503513010a+504\
627200a+522161640a" ]
```

10 The Multiplicity Free Permutation Characters of $G_2(3)$

We compute the multiplicity free possible permutation characters of $G_2(3)$ (see [CCN+85, p. 60]).

For each divisor d of the group order, we compute all those possible permutation characters of degree d of G for which each irreducible constituent occurs with multiplicity at most 1; this is done by prescribing the `maxmult` component of the second argument of `PermChars` to be the list with 1 at each position.

```
gap> t:= CharacterTable( "G2(3)" );
CharacterTable( "G2(3)" )
gap> t:= CharacterTable( "G2(3)" );;
gap> n:= Length( RationalizedMat( Irr( t ) ) );;
gap> maxmult:= List( [ 1 .. n ], i -> 1 );;
gap> perms:= [];;
gap> divs:= DivisorsInt( Size( t ) );;
gap> for d in divs do
>      Append( perms,
>              PermChars( t, rec( bounds  := false,
>                                 degree  := d,
>                                 maxmult := maxmult ) ) );
>    od;
```

```
gap> Length( perms );
42
gap> List( perms, Degree );
[ 1, 351, 351, 364, 364, 378, 378, 546, 546, 546, 546, 546, 702, 702, 728,
  728, 1092, 1092, 1092, 1092, 1092, 1092, 1092, 1092, 1456, 1456, 1638,
  1638, 2184, 2184, 2457, 2457, 2457, 2457, 3159, 3276, 3276, 3276, 3276,
  4368, 6552, 6552 ]
```

For finding out which of these candidates are really permutation characters, we could inspect them piece by piece, using the information in [CCN+85]. For example, the candidates of degrees 351, 364, and 378 are induced from the trivial characters of maximal subgroups of G, whereas the candidates of degree 546 are not permutation characters.

Since the table of marks of G is available in GAP, we can extract all permutation characters from the table of marks, and then filter out the multiplicity free ones.

```
gap> tom:= TableOfMarks( "G2(3)" );
TableOfMarks( "G2(3)" )
gap> tbl:= CharacterTable( "G2(3)" );
CharacterTable( "G2(3)" )
gap> permstom:= PermCharsTom( tbl, tom );;
gap> Length( permstom );
433
gap> multfree:= Intersection( perms, permstom );;
gap> Length( multfree );
15
gap> List( multfree, Degree );
[ 1, 351, 351, 364, 364, 378, 378, 702, 702, 728, 728, 1092, 1092, 2184, 2184
 ]
```

11 Degree 11 200 Permutation Characters of $O_8^+(2)$

We compute the primitive permutation characters of degree 11 200 of $O_8^+(2)$ and $O_8^+(2).2$ (see [CCN+85, p. 85]). The character table of the maximal subgroup of type $3^4 : 2^3.S_4$ in $O_8^+(2)$ is not available in the GAP table library. But the group extends to a wreath product of S_3 and S_4 in the group $O_8^+(2).2$, and the table of this wreath product can be constructed easily.

```
gap> tbl2:= CharacterTable("O8+(2).2");;
gap> s3:= CharacterTable( "Symmetric", 3 );;
gap> s:= CharacterTableWreathSymmetric( s3, 4 );
CharacterTable( "Sym(3)wrS4" )
```

The permutation character `pi` of $O_8^+(2).2$ can thus be computed by explicit induction, and the character of $O_8^+(2)$ is obtained by restriction of `pi`.

```
gap> fus:= PossibleClassFusions( s, tbl2 );
[ [ 1, 41, 6, 3, 48, 9, 42, 19, 51, 8, 5, 50, 24, 49, 7, 2, 44, 22, 42, 12,
      53, 17, 58, 21, 5, 47, 26, 50, 37, 52, 23, 60, 18, 4, 46, 25, 14, 61,
```

```
      20, 9, 53, 30, 51, 26, 64, 8, 52, 31, 13, 56, 38 ] ]
gap> pi:= Induced( s, tbl2, [ TrivialCharacter( s ) ], fus[1] )[1];
Character( CharacterTable( "O8+(2).2" ), [ 11200, 256, 160, 160, 80, 40, 40,
  76, 13, 0, 0, 8, 8, 4, 0, 0, 16, 16, 4, 4, 4, 1, 1, 1, 1, 5, 0, 0, 0, 1, 1,
  0, 0, 0, 0, 0, 2, 2, 0, 0, 1120, 96, 0, 16, 0, 16, 8, 10, 4, 6, 7, 12, 3,
  0, 0, 2, 0, 4, 0, 1, 1, 0, 0, 1, 0, 0, 0 ] )
gap> PermCharInfo( tbl2, pi ).ATLAS;
[ "1a+84a+168a+175a+300a+700c+972a+1400a+3200a+4200b" ]
gap> tbl:= CharacterTable( "O8+(2)" );
CharacterTable( "O8+(2)" )
gap> rest:= RestrictedClassFunction( pi, tbl );
Character( CharacterTable( "O8+(2)" ), [ 11200, 256, 160, 160, 160, 80, 40,
  40, 40, 76, 13, 0, 0, 8, 8, 8, 4, 0, 0, 0, 16, 16, 16, 4, 4, 4, 4, 1, 1, 1,
  1, 1, 1, 5, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 0, 0, 0 ] )
gap> PermCharInfo( tbl, rest ).ATLAS;
[ "1a+84abc+175a+300a+700bcd+972a+3200a+4200a" ]
```

12 A Proof of Nonexistence of a Certain Subgroup

We prove that the sporadic simple Mathieu group $G = M_{22}$ (see [CCN+85, p. 39]) has no subgroup of index 56. In [Isa76], remark after Theorem 5.18, this is stated as an example of the case that a character may be a possible permutation character but not a permutation character.

Let us consider the possible permutation character of degree 56 of G.

```
gap> tbl:= CharacterTable( "M22" );
CharacterTable( "M22" )
gap> perms:= PermChars( tbl, rec( torso:= [ 56 ] ) );
[ Character( CharacterTable( "M22" ), [ 56, 8, 2, 4, 0, 1, 2, 0, 0, 2, 1, 1
     ] ) ]
gap> pi:= perms[1];;
gap> Norm( pi );
2
gap> Display( tbl, rec( chars:= perms ) );
M22

     2  7  7  2  5  4  .  2  .  .  3   .   .
     3  2  1  2  .  .  .  1  .  .  .   .   .
     5  1  .  .  .  .  1  .  .  .  .   .   .
     7  1  .  .  .  .  .  .  1  1  .   .   .
    11  1  .  .  .  .  .  .  .  .  .   1   1

       1a 2a 3a 4a 4b 5a 6a 7a 7b 8a 11a 11b
    2P 1a 1a 3a 2a 2a 5a 3a 7a 7b 4a 11b 11a
    3P 1a 2a 1a 4a 4b 5a 2a 7b 7a 8a 11a 11b
    5P 1a 2a 3a 4a 4b 1a 6a 7b 7a 8a 11a 11b
    7P 1a 2a 3a 4a 4b 5a 6a 1a 1a 8a 11b 11a
   11P 1a 2a 3a 4a 4b 5a 6a 7a 7b 8a  1a  1a
```

```
Y.1    56  8  2  4  .  1  2  .  .  2   1   1
```

Suppose that pi is a permutation character of G. Since G is 2-transitive on the 56 cosets of the point stabilizer S, this stabilizer is transitive on 55 points, and thus G has a subgroup U of index $56 \cdot 55 = 3080$. We compute the possible permutation character of this degree.

```
gap> perms:= PermChars( tbl, rec( torso:= [ 56 * 55 ] ) );;
gap> Length( perms );
16
```

U is contained in S, so only those candidates must be considered that vanish on all classes where pi vanishes. Furthermore, the index of U in S is odd, so the Sylow 2 subgroups of U and S are isomorphic; S contains elements of order 8, hence also U does.

```
gap> OrdersClassRepresentatives( tbl );
[ 1, 2, 3, 4, 4, 5, 6, 7, 7, 8, 11, 11 ]
gap> perms:= Filtered( perms, x -> x[5] = 0 and x[10] <> 0 );
[ Character( CharacterTable( "M22" ), [ 3080, 56, 2, 12, 0, 0, 2, 0, 0, 2, 0,
      0 ] ), Character( CharacterTable( "M22" ),
    [ 3080, 8, 2, 8, 0, 0, 2, 0, 0, 4, 0, 0 ] ),
  Character( CharacterTable( "M22" ), [ 3080, 24, 11, 4, 0, 0, 3, 0, 0, 2, 0,
      0 ] ), Character( CharacterTable( "M22" ),
    [ 3080, 24, 20, 4, 0, 0, 0, 0, 0, 2, 0, 0 ] ) ]
```

For getting an overview of the distribution of the elements of U to the conjugacy classes of G, we use the output of PermCharInfo.

```
gap> infoperms:= PermCharInfo( tbl, perms );;
gap> Display( tbl, infoperms.display );
M22

      2    7  7  2  5  2  3
      3    2  1  2  .  1  .
      5    1  .  .  .  .  .
      7    1  .  .  .  .  .
     11    1  .  .  .  .  .

          1a 2a 3a 4a 6a 8a
     2P   1a 1a 3a 2a 3a 4a
     3P   1a 2a 1a 4a 2a 8a
     5P   1a 2a 3a 4a 6a 8a
     7P   1a 2a 3a 4a 6a 8a
    11P   1a 2a 3a 4a 6a 8a

I.1     3080 56  2 12  2  2
I.2        1 21  8 54 24 36
I.3        1  3  4  9 12 18
I.4     3080  8  2  8  2  4
```

```
I.5          1  3   8 36 24 72
I.6          1  3   4  9 12 18
I.7       3080 24  11  4  3  2
I.8          1  9  44 18 36 36
I.9          1  3   4  9 12 18
I.10      3080 24  20  4  .  2
I.11         1  9  80 18  . 36
I.12         1  3   4  9 12 18
```

We have four candidates. For each the above list shows first the character values, then the cardinality of the intersection of U with the classes, and then lower bounds for the lengths of U-conjugacy classes of these elements. Only those classes of G are shown that contain elements of U for at least one of the characters.

If the first two candidates are permutation characters corresponding to U then U contains exactly 8 elements of order 3 and thus U has a normal Sylow 3 subgroup P. But the order of $N_G(P)$ is bounded by 72, which can be shown as follows. The only elements in G with centralizer order divisible by 9 are of order 1 or 3, so P is self-centralizing in G. The factor $N_G(P)/C_G(P)$ is isomorphic with a subgroup of $\mathrm{Aut}(G) \cong GL(2,3)$ which has order divisible by 16, hence the order of $N_G(P)$ divides 144. Now note that $[G : N_G(P)] \equiv 1 \pmod 3$ by Sylow's Theorem, and $|G|/144 = 3\,080 \equiv -1 \pmod 3$. Thus the first two candidates are not permutation characters.

If the last two candidates are permutation characters corresponding to U then U has self-normalizing Sylow subgroups. This is because the index of a Sylow 2 normalizer in G is odd and divides 9, and if it is smaller than 9 then U contains at most $3 \cdot 15 + 1$ elements of 2 power order; the index of a Sylow 3 normalizer in G is congruent to 1 modulo 3 and divides 16, and if it is smaller than 16 then U contains at most $4 \cdot 8$ elements of order 3.

But since U is solvable and not a p-group, not all its Sylow subgroups can be self-normalizing; note that U has a proper normal subgroup N containing a Sylow p subgroup P of U for a prime divisor p of $|U|$, and $U = N \cdot N_U(P)$ holds by the Frattini argument (see [Hup67, Satz I.7.8]).

13 A Permutation Character of the Lyons group

Let G be a maximal subgroup with structure $3^{2+4} : 2A_5.D_8$ in the sporadic simple Lyons group Ly. We want to compute the permutation character 1_G^{Ly}. (This construction has been explained in [BP98, Section 4.2], without showing explicit GAP code.)

In the representation of Ly as automorphism group of the rank 5 graph `B` with 9 606 125 points (see [CCN+85, p. 174]), G is the stabilizer of an edge. A group S with structure $3.McL.2$ is the point stabilizer. So the two point stabilizer $U = S \cap G$ is a subgroup of index 2 in G. The index of U in S is 15 400, and according to the list of maximal subgroups of $McL.2$ (see [CCN+85, p. 100]), the group U is isomorphic to the preimage in $3.McL.2$ of a subgroup H of $McL.2$ with structure $3_+^{1+4} : 4S_5$.

Using the improved combinatorial method described in [BP98, Section 3.2], all possible permutation characters of degree 15 400 for the group McL are computed. (The method of [BP98, Section 3.3] is slower but also needs only a few seconds.)

```
gap> ly:= CharacterTable( "Ly" );;
gap> mcl:= CharacterTable( "McL" );;
gap> mcl2:= CharacterTable( "McL.2" );;
gap> 3mcl2:= CharacterTable( "3.McL.2" );;
```

```
gap> perms:= PermChars( mcl, rec( degree:= 15400 ) );
[ Character( CharacterTable( "McL" ), [ 15400, 56, 91, 10, 12, 25, 0, 11, 2,
      0, 0, 2, 1, 1, 1, 0, 0, 3, 0, 0, 1, 1, 1, 1 ] ),
  Character( CharacterTable( "McL" ), [ 15400, 280, 10, 37, 20, 0, 5, 10, 1,
      0, 0, 2, 1, 1, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0 ] ) ]
```

We get two characters, corresponding to the two classes of maximal subgroups of index 15 400 in McL. The permutation character $\pi = 1_{H \cap McL}^{McL}$ is the one with nonzero value on the class `10A`, since the subgroup of structure $2S_5$ in $H \cap McL$ contains elements of order 10.

```
gap> ord10:= Filtered( [ 1 .. NrConjugacyClasses( mcl ) ],
>                      i -> OrdersClassRepresentatives( mcl )[i] = 10 );
[ 15 ]
gap> List( perms, pi -> pi[ ord10[1] ] );
[ 1, 0 ]
gap> pi:= perms[1];
Character( CharacterTable( "McL" ), [ 15400, 56, 91, 10, 12, 25, 0, 11, 2, 0,
  0, 2, 1, 1, 1, 0, 0, 3, 0, 0, 1, 1, 1, 1 ] )
```

The character $1_H^{McL.2}$ is an extension of π, so we can use the method of [BP98, Section 3.3] to compute all possible permutation characters for the group $McL.2$ that have the values of π on the classes of McL. We find that the extension of π to a permutation character of $McL.2$ is unique. Regarded as a character of $3.McL.2$, this character is equal to 1_U^S.

```
gap> map:= InverseMap( GetFusionMap( mcl, mcl2 ) );
[ 1, 2, 3, 4, 5, 6, 7, 8, 9, [ 10, 11 ], 12, [ 13, 14 ], 15, 16, 17, 18,
  [ 19, 20 ], [ 21, 22 ], [ 23, 24 ] ]
gap> torso:= CompositionMaps( pi, map );
[ 15400, 56, 91, 10, 12, 25, 0, 11, 2, 0, 2, 1, 1, 0, 0, 3, 0, 1, 1 ]
gap> perms:= PermChars( mcl2, rec( torso:= torso ) );
[ Character( CharacterTable( "McL.2" ), [ 15400, 56, 91, 10, 12, 25, 0, 11,
      2, 0, 2, 1, 1, 0, 0, 3, 0, 1, 1, 110, 26, 2, 4, 0, 0, 5, 2, 1, 1, 0, 0,
      1, 1 ] ) ]
gap> pi:= Inflated( perms[1], 3mcl2 );
Character( CharacterTable( "3.McL.2" ), [ 15400, 15400, 56, 56, 91, 91, 10,
  12, 12, 25, 25, 0, 0, 11, 11, 2, 2, 0, 0, 0, 2, 2, 1, 1, 1, 0, 0, 0, 0, 3,
  3, 0, 0, 0, 1, 1, 1, 1, 1, 1, 110, 26, 2, 4, 0, 0, 5, 2, 1, 1, 0, 0, 1, 1
 ] )
```

The fusion of conjugacy classes of S in Ly can be computed from the character tables of S and Ly given in [CCN+85], it is unique up to Galois automorphisms of the table of Ly.

```
gap> fus:= PossibleClassFusions( 3mcl2, ly );;  Length( fus );
4
gap> g:= AutomorphismsOfTable( ly );;
gap> OrbitLengths( g, fus, OnTuples );
[ 4 ]
```

Now we can induce 1_U^S to Ly, which yields $(1_U^S)^{Ly} = 1_U^{Ly}$.

```
gap> pi:= Induced( 3mcl2, ly, [ pi ], fus[1] )[1];
Character( CharacterTable( "Ly" ), [ 147934325000, 286440, 1416800, 1082,
  784, 12500, 0, 672, 42, 24, 0, 40, 0, 2, 20, 0, 0, 0, 64, 10, 0, 50, 2, 0,
  0, 4, 0, 0, 0, 0, 4, 0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0 ] )
```

All elements of odd order in G are contained in U, for such an element g we have

$$1_G^{Ly}(g) = \frac{|C_{Ly}(g)|}{|G|} \cdot |G \cap Cl_{Ly}(g)| = \frac{|C_{Ly}(g)|}{2 \cdot |U|} \cdot |U \cap Cl_{Ly}(g)| = \frac{1}{2} \cdot 1_U^{Ly}(g) \ ,$$

so we can prescribe the values of 1_G^{Ly} on all classes of odd element order. For elements g of even order we have the weaker condition $U \cap Cl_{Ly}(g) \subseteq G \cap Cl_{Ly}(g)$ and thus $1_G^{Ly}(g) \geq \frac{1}{2} \cdot 1_U^{Ly}(g)$, which gives lower bounds for the value of 1_G^{Ly} on the remaining classes.

```
gap> orders:= OrdersClassRepresentatives( ly );
[ 1, 2, 3, 3, 4, 5, 5, 6, 6, 6, 7, 8, 8, 9, 10, 10, 11, 11, 12, 12, 14, 15,
  15, 15, 18, 20, 21, 21, 22, 22, 24, 24, 24, 25, 28, 30, 30, 31, 31, 31, 31,
  31, 33, 33, 37, 37, 40, 40, 42, 42, 67, 67, 67 ]
gap> torso:= [];;
gap> for i in [ 1 .. Length( orders ) ] do
>      if orders[i] mod 2 = 1 then
>        torso[i]:= pi[i]/2;
>      fi;
>    od;
gap> torso;
[ 73967162500,, 708400, 541,, 6250, 0,,,, 0,,, 1,,, 0, 0,,,, 25, 1, 0,,, 0, 0,
  ,,,,, 0,,,, 0, 0, 0, 0, 0, 0, 0, 0, 0,,,,, 0, 0, 0 ]
```

Exactly one possible permutation character of Ly satisfies these conditions.

```
gap> perms:= PermChars( ly, rec( torso:= torso ) );;
gap> Length( perms );
43
gap> perms:= Filtered( perms, cand -> ForAll( [ 1 .. Length( orders ) ],
>        i -> cand[i] >= pi[i] / 2 ) );
[ Character( CharacterTable( "Ly" ), [ 73967162500, 204820, 708400, 541, 392,
      6250, 0, 1456, 61, 25, 0, 22, 10, 1, 10, 0, 0, 0, 32, 5, 0, 25, 1, 0,
      1, 2, 0, 0, 0, 0, 4, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2,
      0, 0, 0, 0, 0 ] ) ]
```

(The permutation character 1_G^{Ly} was used in the proof that the character χ_{37} of Ly (see [CCN+85, p. 175]) occurs with multiplicity at least 2 in each character of Ly that is induced from a proper subgroup of Ly.)

References

[BP98] Thomas Breuer and Götz Pfeiffer, *Finding Possible Permutation Characters*, J. Symbolic Comput. **26** (1998), 343–354.

[CCN+85] J[ohn] H. Conway, R[obert] T. Curtis, S[imon] P. Norton, R[ichard] A. Parker, and R[obert] A. Wilson, *Atlas of finite groups*, Oxford University Press, 1985.

[Hup67] B[ertram] Huppert, *Endliche Gruppen I*, Grundlehren Math. Wiss., vol. 134, Springer, Berlin, Heidelberg and New York, 1967.

[Isa76] I. M. Isaacs, *Character theory of finite groups*, Pure and applied mathematics, vol. 69, Academic Press, New York, 1976, xii+303 pp., ISBN 0-12-374550-0.