Modular Forms Project: Zero Fourier Coefficients of Eta Quotients

Henri Cohen,
Université de Bordeaux, Institut de Mathématiques de Bordeaux,
351 Cours de la Libération, 33405 TALENCE Cedex, FRANCE

July 8, 2018

1 Answers and Programs

Exercise 0.

(1). Let $a/c \in P_1(\mathbb{Q})$ be a cusp with $\gcd(a,c)=1$, let b and d be arbitrary integers such that $ad-bc=1$, and let $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma$. If F is a meromorphic function which is modular on $\mathfrak{H}$ for some subgroup of Γ, the order of vanishing of F at a/c is by definition the order of vanishing of $F|_k\gamma$ at infinity.

Fix some integer m, let $g=\gcd(am,c)=\gcd(m,c)$, $A=am/g$, $C=c/g$, let B and D be arbitrary such that $AD-BC=1$, and set $h=mbD-dB$. We have the following matrix equality:

$$\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}\begin{pmatrix} g & h \\ 0 & m/g \end{pmatrix}\;,$$

which we abbreviate into $\beta_m\gamma = \delta\left(\begin{smallmatrix} g & h \\ 0 & m/g \end{smallmatrix}\right)$.

(2). Since $\eta|_{1/2}\beta_m = m^{1/4}\eta(m\tau)$, we thus have

$$\eta(m\tau)|_{1/2}\gamma = m^{-1/4}\eta|_{1/2}\beta_m\gamma = \zeta m^{-1/4}\eta_{1/2}\left(\begin{smallmatrix} g & h \\ 0 & m/g \end{smallmatrix}\right)$$

for some 24th root of unity ζ since η is modular on Γ. Thus

$$\eta(m\tau)|_{1/2}\gamma = \zeta(m/g)^{-1/2}\eta((g^2\tau+gh)/m)\;,$$

and the order of vanishing at infinity of this expression is clearly equal to $g^2/(24m) = \gcd(m,c)^2/(24m)$.

(3). Thus, the order of vanishing of $F_m(r,s)$ at the cusp a/c is equal to $(r+s\gcd(m,c)^2/m)/24$, so F_m is holomorphic if and only if for all c we have $r+s\gcd(m,c)^2/m \ge 0$.

Now $1 \le \gcd(m,c) \le m$, and both ends of the interval are trivially attained (for $c=1$ and $c=m$), so the above condition reduces to $r+s/m \ge 0$ and $r+sm \ge 0$, i.e., $r \ge \max(-s/m, -sm)$.

Experiment 1.

Even for $r \leq 50$ and $X = 5 \cdot 10^7$ I did not find any zero Fourier coefficient.

Experiment 2.

(1). We simply write the following naive program:

```
/* Given a power series F with L terms, output
   vector of zeros. */
findzeros(F,L)=
{
  my(ct=0,V);
  for(n=0,L,if(polcoeff(F,n)==0,ct++));
  V=vector(ct); ct=0;
  for(n=0,L,if(polcoeff(F,n)==0,ct++;V[ct]=n));
  return(V);
}
L=10^6; /* On a machine with at least 32 GO of main memory
         one can go to L=10^8. I only went to 5.10^7. */
E=eta(x+O(x^(L+1)));
E2=E^2; /* Will be needed later. */
E3=E*E2; /* Very lacunary, so useful. */
E5=E3*E2;
V=findzeros(E5,L); V5=apply(x->24*x+5,V);
```

The first entries of `V5` are

[37445, 43253, 44117, 55637, 64565, 76181, 89813, 94205, 104357, 113045, ...]

To check whether they are really Shimura zeros, and in fact isolate the initial values, we write the following:

```
L5=24*L+5; W=V5;
for(i=1,#W,n=W[i];
  if(n,
    forstep(m=5,sqrtint(L5\n),[2,4],
      r=vecsearch(V5,n*m^2);
      if(!r,print1([i,m]," ");error("not a Shimura zero?"));W[r]=0
    )
  )
);
V5S=vecsort(Vec(Set(W))); V5S=V5S[2..#V5S];
#V5S
```

Don't worry about the one-to-last line. First, since the program does not output an error, we indeed have only Shimura zeros (up to our limit), and the one-to-last line simply removes the zeros that we have added.

If you chose $L = 10^6$, you will find 347 zeros, corresponding to 235 Shimura zeros.

I went up to $L = 5 \cdot 10^7$ and found 2926 zeros, corresponding to 1080 Shimura zeros.

(2).

```
E7=E5*E2; V=findzeros(E7,L); V7=apply(x->24*x+7,V);
```

For $L = 10^6$ we find two zeros `V7=[672415, 16810375]`, and the second component is indeed 25 times the first, so it looks like a Shimura zero, so to be sure we have to go further, and indeed this is confirmed, as is the fact that 672415 is the only Shimura zero I found up to $L = 5 \cdot 10^7$ (corresponding to $24L + 7 > 10^9$).

(3).

```
E15=E3^5; /* or anything else. */
V=findzeros(E15,L); V15=apply(x->8*x+5,V);
```

This gives 68 zeros, the first being 429, but all the other zeros (including up to $5 \cdot 10^7$) are of the form $429k^2$, so once again only a single Shimura zero found.

(4). As mentioned, no other Shimura zeros found up to $L = 5 \cdot 10^7$ and r odd with $r \leq 43$.

Although not needed or asked for, we comment on the use of the explicit formula for η^5. For $x + iy \in \mathbb{Z}[i]$ coprime to 3 and to $1 + i$, define $\psi(x + iy) = i^m \left(\frac{-4}{x^2-y^2}\right)$, where m is the unique integer modulo 4 such that $(x+iy)(1-i)^m = a + ib$ with $3 \mid b$.

Lemma 1.1 *This makes sense, ψ is multiplicative, and depends only on the ideal $(x + iy)\mathbb{Z}[i]$. In other words ψ is a* Hecke character *of finite order.*

Proof. We have

$$(x_1x_2-y_1y_2)^2-(x_1y^2+y_1x_2)^2 = (x_1^2-y_1^2)(x_2^2-y_2^2)-4x_1x_2y_1y_2 \equiv (x_1^2-y_1^2)(x_2^2-y_2^2) \pmod 4 ,$$

so $\left(\frac{-4}{x^2-y^2}\right)$ is multiplicative. Furthermore

$$(1 - i)^m \equiv \pm 1, \pm(1 - i), \pm i, \pm(1 + i) \pmod 3\mathbb{Z}[i]$$

when $m \equiv 0, 1, 2, 3 \pmod 4$, so

$$(x+iy)(1-i)^m \equiv \pm(x+iy), \pm(x+y+i(y-x)), \pm(-y+ix), \pm(x-y+i(x+y)) \pmod{3\mathbb{Z}[i]} ,$$

and since x or y is not divisible by 3, exactly one of y, $y - x$, x, or $y + x$ is divisible by 3, proving existence and uniqueness of m modulo 4, and clearly m is additive, so i^m is multiplicative. Finally, multiplying $x + iy$ by i amounts to changing (x, y) into $(-y, x)$, so $\left(\frac{-4}{x^2-y^2}\right)$ changes sign, and it is immediate to see that m is changed into $m + 2$ modulo 4, so $\psi(-y + ix) = \psi(x + iy)$. □

It follows from the last part of this lemma that if $\mathfrak{a}$ is an ideal of $\mathbb{Z}[i]$ coprime to 3 we can set $\psi(\mathfrak{a}) = \psi(x + iy)$ for any generator $x + iy$ of $\mathfrak{a}$.

Corollary 1.2 *If $x^2 + y^2 \equiv 1 \pmod 3$ we have $\psi(x + iy) = \left(\frac{12}{x^2-y^2}\right)$, and if $x^2 + y^2 \equiv 2 \pmod 3$ we have $\psi(x + iy) = \pm i\left(\frac{-4}{x^2-y^2}\right)$ for a suitable sign $\pm$.*

Proof. Immediate from the lemma by inspection. □

Corollary 1.3 *For $\gcd(n, 6) = 1$, let $c(n) = \sum_{\mathcal{N}(\mathfrak{a})=n} \psi(\mathfrak{a})$. Then $c(n)$ is multiplicative, and for $n = p^a$ we have the following:*

(1) *If* $p \equiv 3 \pmod 4$ *we have* $c(p^a) = 0$ *if* a *is odd and* $c(p^a) = 1$ *if* a *is even.*

(2) *If* $p \equiv 5 \pmod{12}$ *we have* $c(p^a) = 0$ *if* a *is odd and* $c(p^a) = (-1)^{a/2}$ *if* a *is even.*

(3) *If* $p \equiv 1 \pmod{12}$ *we have* $c(p^a) = s^a(a+1)$*, where* $s = 1$ *if* p *is of the form* $x^2 + 36y^2$ *and* $s = -1$ *otherwise.*

Proof. Multiplicativity follows from the lemma. Thus, assume $n = p^a$ with $p > 3$.

If $p \equiv 3 \pmod 4$, we have $c(p^a) = 0$ if a is odd, and otherwise since p is inert we have $\mathfrak{a} = p^{a/2}\mathbb{Z}[i]$, and since $p^a \equiv 1 \pmod 4$ by the corollary we have $\psi(\mathfrak{a}) = \left(\frac{12}{p^a}\right) = 1$, so $c(n) = 1$.

If $p \equiv 1 \pmod 4$ with $3 \nmid p$ (i.e., $p \equiv 1, 5 \pmod{12}$), let $\mathfrak{p}$ be above p and $x + iy$ be any generator of $\mathfrak{p}$. The ideals of norm p^a are $\mathfrak{p}^r\overline{\mathfrak{p}}^{a-r}$ for $0 \le r \le a$, so

$$c(n) = \sum_{0 \le r \le a} \psi(\mathfrak{p})^r \psi(\overline{\mathfrak{p}})^{a-r} .$$

Again by the corollary, since $p^2 \equiv 1 \pmod 3$ we have $\psi(p) = \left(\frac{12}{p^2}\right) = 1$, hence $\psi(\overline{\mathfrak{p}}) = \overline{\psi(\mathfrak{p})}$.

Assume first that $p \equiv 5 \pmod{12}$, so that if $x^2 + y^2 = p$ we have $3 \nmid xy$. It is then clear that m is odd, so $\psi(\mathfrak{p}) = i\varepsilon$ for some $\varepsilon = \pm 1$, hence $\psi(\overline{\mathfrak{p}}) = \overline{\psi(\mathfrak{p})} = -i\varepsilon$. Thus

$$c(n) = i^a \sum_{0 \le r \le a} \varepsilon^r (-\varepsilon)^{a-r} = (-i\varepsilon)^a \sum_{0 \le r \le a} (-1)^r .$$

If a is odd, we thus have again $c(n) = 0$, while if a is even we have $c(n) = (-1)^{a/2}$, as claimed.

Assume finally that $p \equiv 1 \pmod{12}$. By the corollary we have $s := \psi(\mathfrak{p}) = \psi(\overline{\mathfrak{p}}) = \left(\frac{12}{x^2 - y^2}\right)$, hence $c(n) = s^a(a+1)$. Since $12 > 0$ we may exchange x and y if necessary so that $2 \mid y$, hence $2 \nmid x$. If $3 \mid y$, in other words if p is of the form $x^2 + 36y^2$, we have x coprime to 6 so $x^2 - y^2 \equiv 1 \pmod{12}$ hence $s = 1$. On the other hand, if $3 \nmid y$ we have $y^2 \equiv 0 \pmod 4$ and $1 \pmod 3$, hence $4 \pmod{12}$, and we must have $3 \mid x$ hence $x^2 \equiv 1 \pmod 4$ and $0 \pmod 3$, hence $9 \pmod{12}$, so $x^2 - y^2 \equiv 7 \pmod{12}$ so $s = -1$. $\square$

Set $\eta^5(\tau) = \sum_{n \equiv 5 \pmod{24}} e_5(n) q^{n/24}$. By Proposition 1.8, we have

$$e_5(n) = (1/8) \sum_{x^2+y^2+3z^2=n} \left(\frac{12}{xy}\right)\left(\frac{-4}{z}\right) z = (1/4) \sum_{\substack{1 \le z \le (n/3)^{1/2} \\ 2 \nmid z}} (-1)^{(z-1)/2} z \sum_{x^2+y^2=n-3z^2} \left(\frac{12}{xy}\right) ,$$

where $n - 3z^2 \equiv 2 \pmod{24}$. The inner sum vanishes unless x and y are both odd, so settting $X = (x+y)/2$ and $Y = (x-y)/2$ we have

$$e_5(n) = (1/4) \sum_{\substack{1 \le z \le (n/3)^{1/2} \\ 2 \nmid z}} (-1)^{(z-1)/2} z f((n - 3z^2)/2) ,$$

where for $n \equiv 1 \pmod{12}$ we have

$$f(n) = \sum_{X^2+Y^2=n} \left(\frac{12}{X^2-Y^2}\right) = \sum_{X^2+Y^2=n} \psi(X+iY) = 4 \sum_{\mathcal{N}(\mathfrak{a})=n} \psi(\mathfrak{a}) = 4c(n)$$

by Corollary 1.2.

We can thus write the following programs to compute $f(n)$, hence also $e_2(n)$ and $e_5(n)$:

```
global(Q=Qfb(1,0,36));
/* Given $n\equiv1\pmod{12}$ (not checked), compute c(n).
Q is the fixed quadratic form x^2+36y^2 given by Qfb(1,0,36). */
fun0(n)=
{
  my(fa,lipr,liex,R=1,p,a);
  if(n==1,return(1));
  fa=factor(n); lipr=fa[,1]; liex=fa[,2];
  for(i=1,#lipr,
    p=lipr[i]; a=liex[i];
    if (p%4==3,if(a%2,return(0));next());
    if (p%12==5,if(a%2,return(0),if(a%4==2,R=-R));next());
    R*=(a+1);
    if (a%2 && !qfbsolve(Q,p),R=-R);
  );
  R;
}
/* Assume $n\equiv5\pmod{24}$. */
e5(n)=
{
  my(limz=sqrtint(n\3),S=0,T);
  forstep(z=1,limz,2,
    T=z*fun0((n-3*z^2)/2);
    if(z%4==3,S-=T,S+=T)
  );
  S;
}
```

Note that this program is very efficient to compute a *single* Fourier coefficient, but if one wants all the Fourier coefficients up to a given bound it is still preferable to use power series.

Implementation exercise 1.

(1).

On my laptop, the commands

```
for(j=4,7,gettime();eta(x+O(x^(10^j)));print1(gettime()," "));
```

outputs 4 24 600 28581 (which are milliseconds), so already more than 28 seconds for 10^7 terms. We can write the following trivial script:

```
/* Computes the first L coefficients of eta(m\tau), m=1 by default. */
```

```
myeta(L,m=1)=
{
  my(V,s,n,Lm=L\m);
  V=vector(L+1); V[1]=1; s=1;
  for(i=1,oo,
    s=-s; n=i*(3*i-1)/2; if(n<=Lm,V[m*n+1]=s,break());
    n+=i; if(n<=Lm,V[m*n+1]=s,break())
  );
  return (Ser(V));
}
```

and the commands

```
for(j=4,8,gettime();myeta(10^j);print1(gettime()," "));
```

where we even go up to 10^8 outputs `0 4 32 280 2857`, so is orders of magnitude faster. The only reason why I cannot go up to 10^9 is due to memory constraints, not time.

The `Pari` library command `eta_inflate_ZXn` goes another order of magnitude faster, the times being `0 0 4 24 216`.

(2).
For $4 \leq j \leq 5$ I obtain times of 92 and 6905 ms respectively.

(3). For $L = 10^4$, I obtain a time of 4048 ms, orders of magnitude larger than the 92 ms necessary to compute $1/E$ directly.

(4). We first write:

```
install(ser_inv,G);
p=nextprime(10^9);
E=eta(x+O(x^(10^5)));
Ep=Mod(E,p);
#
1/E;
ser_inv(E);
ser_inv(Ep);
```

The respective times are 6.9s (as before), 2.2s using `ser_inv`, already three times faster, but only 0.3s (23 times faster) modulo p. We can also write

```
for(j=4,7,Ep=Mod(eta_inflate_ZXn(10^j,1),p);gettime();ser_inv(Ep);print1(gettime()," "))
```

which outputs `24 296 3609 48152`, so even for $L = 10^7$, only 48s are necessary to invert eta modulo p.

Implementation Exercise 2.
(1). The general program to compute $F_2(r, s)$ to L terms could be as follows:

```
/* First part, to be done once and for all,
independently of $r$ and $s$.*/
install(ser_inv,G);
global(p,L,Ep,Epinv,E2p,E2pinv);
precompute(L0,p0=nextprime(10^9))=
```

```
{
  L=L0; p=p0;
  Ep=Mod(eta_inflate_ZXn(L+1,1),p);
  Epinv=ser_inv(Ep);
  E2p=Mod(eta_inflate_ZXn(L+1,2),p);
  E2pinv=ser_inv(E2p);
}
```

The syntax of this function means that the second argument `p0` is optional, and if not given will be set to `nextprime(10^9)`.

Note that it would seem more logical to obtain `E2p` and `E2pinv` simply by substituting x^2 for x, but the substitution function is very slow, except if written carefully for this specific case.

(2).

Once these precomputations done, the rest is as follows:

```
/* Function F_2(r,s), length L, modulo p, both implicit. */
F2(r,s)=
{
  my(E1,E2);
  E1=if(r<0,Epinv^(-r),Ep^r);
  E2=if(s<0,E2pinv^(-s),E2p^s);
  return(E1*E2);
}
```

(3).

```
/* Find list of zeros in correct format. L and p are implicit. */
findzeros2(r,s)=
{
  my(F,Z,ord,den,num);
  F=F2(r,s); Z=findzeros(F,L);
  ord=(r+2*s)/24;
  den=denominator(ord); num=numerator(ord);
  return(apply(x->den*x+num,Z));
}
```

To test our program without going further, we can first write `precompute(10^4)`, then `findzeros2(5,0)` which outputs the first Shimura zeros of η^5: [37445, 43253,...].

Experiment 3.

Now that these preliminary programs are written, the rest will be the interesting part, i.e., observation. We first write `precompute(10^6);` which only requires 7.4s.

(1). We simply write `V=findzeros2(-8,24)`, which requires 10.8s. To look at the first terms we write `V[1..10]` which outputs $[14, 26, 38, \cdots]$. Indeed, we check that $V[i] = 12i + 2$:

```
for(i=1,#V,if(V[i]!=12*i+2,print(i);error("really?")))
```

Since the valuation at infinity is $40/24 = 5/3$, the density of zeros is exactly $1/4$. It should be easy to prove (at least that these coefficients are zero, perhaps not that there are no others).

(2). Similarly, writing `V=findzeros2(-5,13)` requires 11s, and the first entries of the output are $[31, 279, 527, ...]$. All these values are divisible by 31. More precisely, the valuation at infinity is $21/24 = 7/8$, so the elements of $\mathcal{E}(d)$ are congruent to 7 modulo 8, hence the values can only be of the form $31h$ with $h \equiv 1 \pmod 8$, and indeed writing `V[1..20]/31` outputs $[1, 9, 17, \cdots]$. Thus a hasty conjecture would be that all numbers of the form $31h$ with $h \equiv 1 \pmod 8$ are Fourier zeros. However, if you read the exercise carefully, you will notice that I mention *odd* powers of 31. And indeed, we have $217 \equiv 1 \pmod 8$ (and divisible by 31), but $31 \cdot 217$ does *not* occur as a Fourier zero. On the other hand 31^3 does occur. Thus, a more plausible conjecture is that all numbers $n \in \mathcal{E}(d)$ such that $v_{31}(n)$ is odd occur as Fourier zeros.

If this conjecture is true, the density of such zeros is $1/31 - 1/31^2 + 1/31^3 - \cdots = 1/32$, and I have not found any other.

Exactly the same phenomenon occurs for $(1, 7)$ ($h \equiv 3 \pmod 8$), $(7, 1)$ ($h \equiv 5 \pmod 8$), and $(13, -5)$ ($h \equiv 7 \pmod 8$).

In view of this regularity, I believe this conjecture should not be difficult to prove.

(3). Similarly, we write `V=findzeros2(-3,13)`, which requires 11s, and the first entries of the output are $[1223, 30575, 59927, ...]$. All these values are divisible by 1223. The valuation at infinity is $23/24$, so the elements of $\mathcal{E}(d)$ are congruent to 23 modulo 24, hence the values can only be of the form $1223h$ with $h \equiv 1 \pmod{24}$, and indeed writing `V[1..20]/1223` outputs $[1, 25, 49, 97, ...]$. In view of the previous example, we suspect that only the odd powers of 1223 will occur. To check this, we need to compute the first Fourier coefficient divisible by 1223^2 and congruent to 23 modulo 24. This is for $n = 23 \cdot 1223^2$, corresponding to the mth Fourier coefficient that we have computed with $m = (n - 23)/24 = 1433406$. This is slightly above our default computation with $L = 10^6$, so we must start again with, for example, $L = (3/2) \cdot 10^6$, and indeed, n does not occur (instead of looking "by hand" you can use the `GP` function `vecsearch`). To be really shure of our conjecture, we would need to check that $n = 1223^3$ does occur, but since $m = (n-23)/24 = 76219856$ this would require computing almost $8 \cdot 10^7$ Fourier coefficients, which is just at the limit of what can be done on a laptop (but of course could be tested on a computer with at least $32G$ of main memory).

Once again, if this conjecture is true, the density of such zeros is $1/(1223 + 1) = 1/1224$, and I have not found any other.

Exactly the same phenomenon occurs for $(1, 9)$ ($h \equiv 5 \pmod{24}$), $(9, 1)$ ($h \equiv 13 \pmod{24}$), and $(13, -3)$ ($h \equiv 17 \pmod{24}$).

Once again, in view of this regularity, I believe this conjecture should not be difficult to prove.

(4). As mentioned, I have not found any other nonlacunary $F_2(r, s)$ with congruence zeros in integral weight.

Exercise 1.

(1). We are going to use the full power of the modular forms package of `Pari/GP`. First, as stated, we first write `F=mffrometaquo([8,1;16,7]);mfparams(F);`

which answers [128,4,8,y], which means that $F \in M_4(\Gamma_0(128), \chi_8)$ (but we know that all our eta quotients will be cusp forms), and the y means that the coefficients will be in the field $\mathbb{Q}(\chi_8)$, i.e., here simply rational numbers (in fact integers). The last command mf=mfinit(F,0) initializes the space in whch F belongs, the parameter 0 meaning the new space (we do not know in advance that F is in the new space, but since the command does not give an error, it does).

(2). The command is simply B=mfeigenbasis(mf);. The command mffields(mf) gives

```
[y^2 + 2, y^2 + 1, y^2 + 1, y^2 + 1, y^4 + 4*y^2 + 9]
```

which means that the first two eigenforms are (conjugates) and defined over $\mathbb{Q}(\sqrt{-2})$, the next six are pairwise conjugate and defined over $\mathbb{Q}(i)$, and the last four are Galois conjugates over a quartic field.

We will solve the question in the two ways which are suggested.

(a). First, let us use linear algebra. We need to construct a matrix with sufficiently many rows (since the dimension is 12 let us choose 20), whose columns are the coefficients of the eigenforms, but expressed as complex numbers, and then solve a linear system with complex coefficients. For this, we used the Pari/GP function mfembed as follows:

```
mftoeigenbasis(mf,F)=
{
  my(B,lim,M,tmp);
  B=mfeigenbasis(mf); lim=max(mfsturm(mf),2*mfdim(mf))+1;
  M=Mat([]);
  for(i=1,#B,
    tmp=mfembed(B[i],mfcoefs(B[i],lim));
    for(j=1,#tmp,M=concat(M,Mat(tmp[j]~)))
  );
  matsolve(M,mfcoefs(F,lim)~);
}
```

After execution of this (general) program on our specific mf and F, and suppression of the coefficients which are clearly zero (say less than 10^{-30}), we find as coefficients $C \cdot [0,0,0,0,0,0,0,0,1,-1,-1,1]$, where C is the constant $C = -0.013975424859373685602557335429570476472 * I$ which the command algdep(C,2) immediately recognizes as $1/\sqrt{-5120}$. It follows that in fact our eta quotient F is simply a linear combination of the four Galois conjugates of the last eigenform with simple coefficients.

(b). We now use a more expensive but amusing method: if F_i are the eigenforms, the coefficient of F on F_i is simply $< F, F_i > / < F_i, F_i >$. To compute these Petersson products, we must use the function mfsymbol. We assume already computed mf and B as above.

```
FS=mfsymbol(mf,F);
BS=apply(x->mfsymbol(mf,x),B);
for(i=1,5,
  num=mfpetersson(FS,BS[i]);
```

```
  matden=mfpetersson(BS[i],BS[i]);
  for(j=1,#num,
    print(num[j]/matden[j,j])
  )
)
```

Some comments are in order: `mfpetersson(FS,BS[i])` gives a *vector* of results, one for each embedding of `BS[i]` (since `FS` has only one embedding). However, logically `mfpetersson(BS[i],BS[i])` gives a *matrix* of results, the rows and columns corresponding to each embedding of `BS[i]`. Of course, by orthogonality of eigenforms, this matrix is (approximately) a diagonal matrix.

The output of this program is of course exactly the same as using linear algebra.

(3). We simply use the `mfcoefs` command on `B[5]`: more precisely, `lift(mfcoefs(B[5],31))` outputs

```
[0,1,0,-2/3*y^3-14/3*y,0,8*y^2+16,0,...0,56*y^2+112,0,0]
```

This shows that the coefficients $a(n)$ with $2 \mid n$ are 0, which is equivalent to $T(2)$ having zero eigenvalue, since the level is divisible by 2. But we see that $a(31)$ also vanishes, which means that these last four eigenforms have zero eigenvalue for $T(31)$. However, 31 does not divide the level, hence $T(31)T(31) = T(31^2) + \chi_8(31) \cdot 31^3 T(1)$, in other words $a(31^2) = -31^3$ (which we can check by writing `mfcoef(B[5],31^2)` with no "s"), and by induction of course $a(31^k) = 0$ for k odd, and by multiplicativity $a(m \cdot 31^k) = 0$ for k odd and $31 \nmid m$, proving our conjecture. Of course this does not imply that there are no other zero Fourier coefficients.

Experiment 4.

We want to search on r, s, so we do not want to spend too much time on each one. Before focusing on particular pairs, it is preferable to precompute to a smaller limit, for instance `precompute(10^5)`.

Proposition 1.2 tells us that we must have $r \geq \max(-2s, -s/2)$, in other words when $s > 0$ we must have $r \geq -s/2$, and when $s < 0$ we must have $r \geq -2s$ (we do not need $s = 0$ since this corresponds to pure eta powers which we have already studied). We want an estimate of the zero density: if it is close to 1, the series is probably lacunary, if it is close to 0 the series has probably no congruence zeros. The total number of Fourier coefficients found is equal to L (10^5 here for instance), so the density is simply $|V|/L$, where $|V|$ is the number of elements of the vector V output by `findzeros2`.

However, here we are asked to do a search, and in fact we are going to combine this search with the one asked for in the previous experiment. Computing $F_2(r,s)$ independently for each r, s is wasteful since $F_2(r+1,s) = \eta F_2(r,s)$. Thus, we need to rewrite our `findzeros2` program as follows:

```
/* Find zeros, given F_2(r-1,s), and output F_2(r,s). */
findzeros3(r,s,F2prev=0)=
{
  my(F,Z,ord,den,num);
  if(F2prev,F=Ep*F2prev,F=F2(r,s));
  Z=findzeros(F,L); ord=(r+2*s)/24;
```

```
  den=denominator(ord); num=numerator(ord);
  return([apply(x->den*x+num,Z),F]);
}
```

The above syntax simply means that if F2prev is given (corresponding to $F_2(r-1,s)$ we compute $F_2(r,s)$ by simply multiplying it by eta, and otherwise we compute it from scratch, and in both cases we return $F_2(r,s)$ as second component of the result.

We will do the search for instance for $|s| \leq 35$ and $|r| \leq 50$.

```
densinner(s,r0,limr0=50)=
{
  my(c,F,V,nb);
  F=0;
  for(r=r0,limr0,
    if (!r,next());
    [V,F]=findzeros3(r,s,F); nb=#V;
    if (nb>1,print([r,s],": found ",nb,", ",V[1..min(10,nb)]));
  );
}

density(lims=35,limr0=50)=
{
  my(r0);
  forstep(s=-1,-lims,-1,
    r0=-2*s;
    densinner(s,r0,limr0)
  );
  for(s=1,lims,
    r0=ceil(-s/2);
    densinner(s,r0,limr0)
  );
}
```

This program (after running several hours, less on multiple processors) outputs a large amount of data.