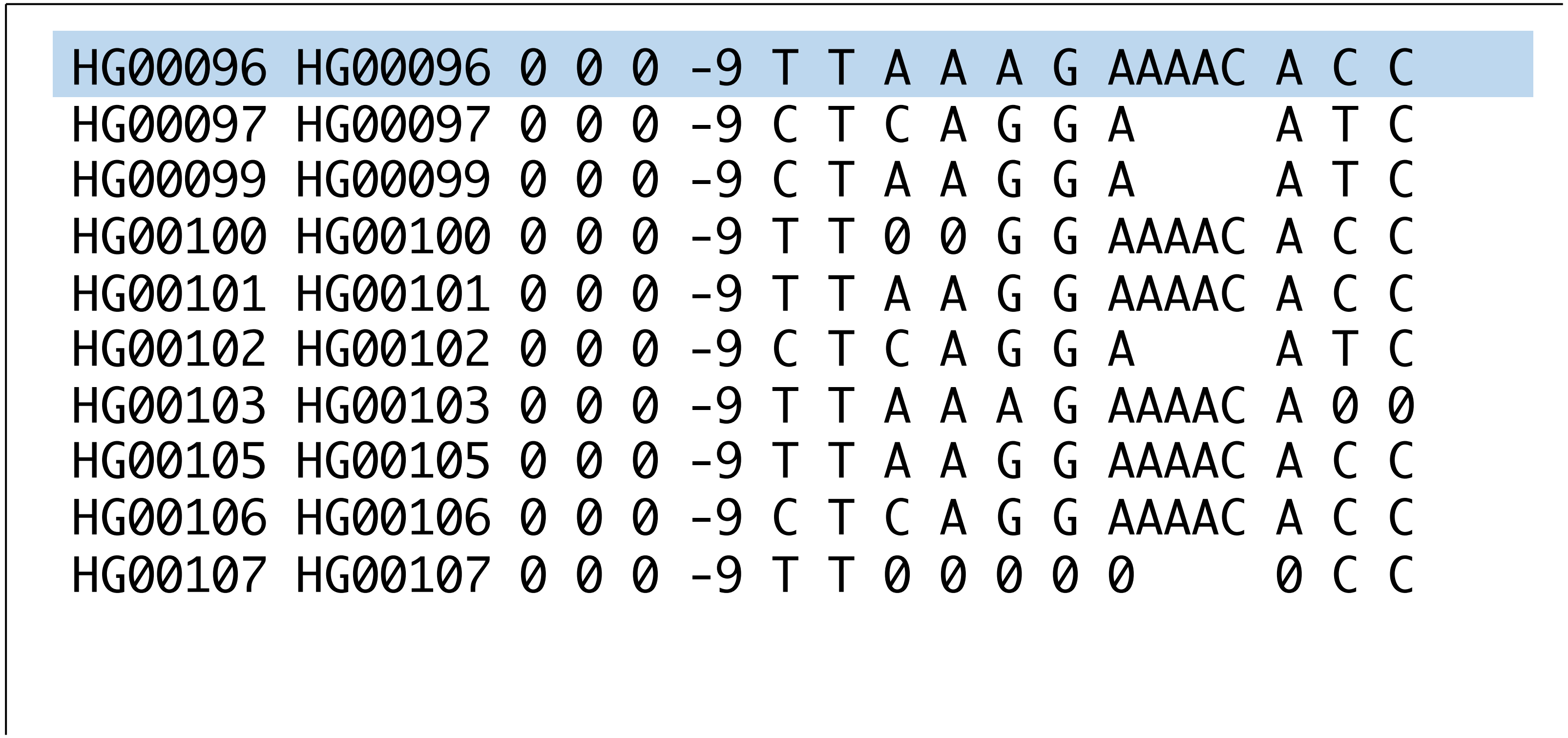
This is a portion of the ped file you are using:

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

It gives genotype data for (number of columns - 6)/2 variants for (number of rows) individuals

Each line contains the data for one individual. Here I’m highlighting the data for individual “HG00096”.



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

And here’s HG00097:

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

The first six columns give identifying information for each individual in the file.

Family ID - in this case, the same as…

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

The first six columns give identifying information for each individual in the file.

Individual ID

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

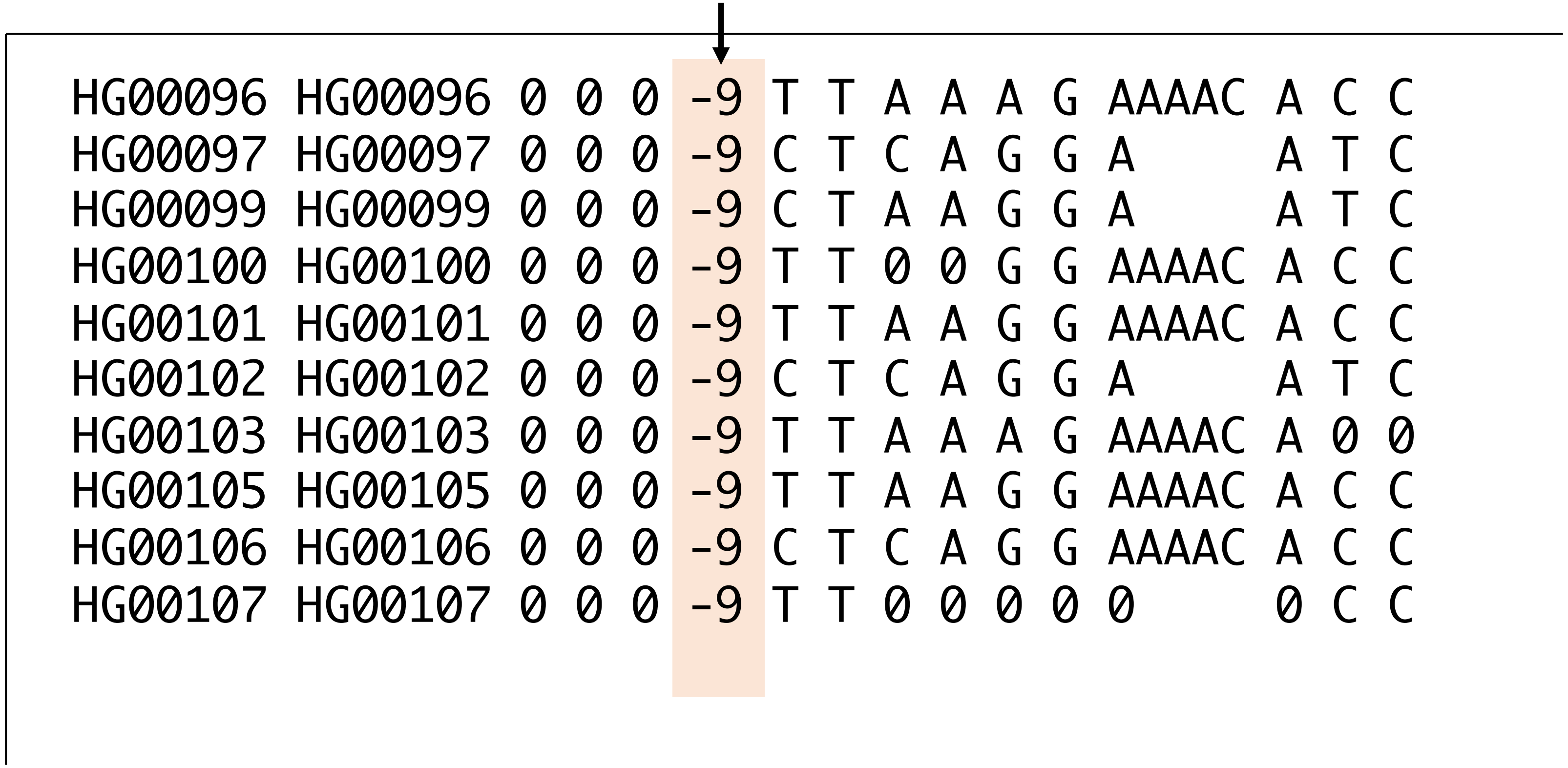
The first six columns give identifying information for each individual in the file.

Usually give ID of father and mother (if present in the dataset) and sex, but missing in our file

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

The first six columns give identifying information for each individual in the file.

“Affection status” or phenotype; we don’t care about this column for this assignment

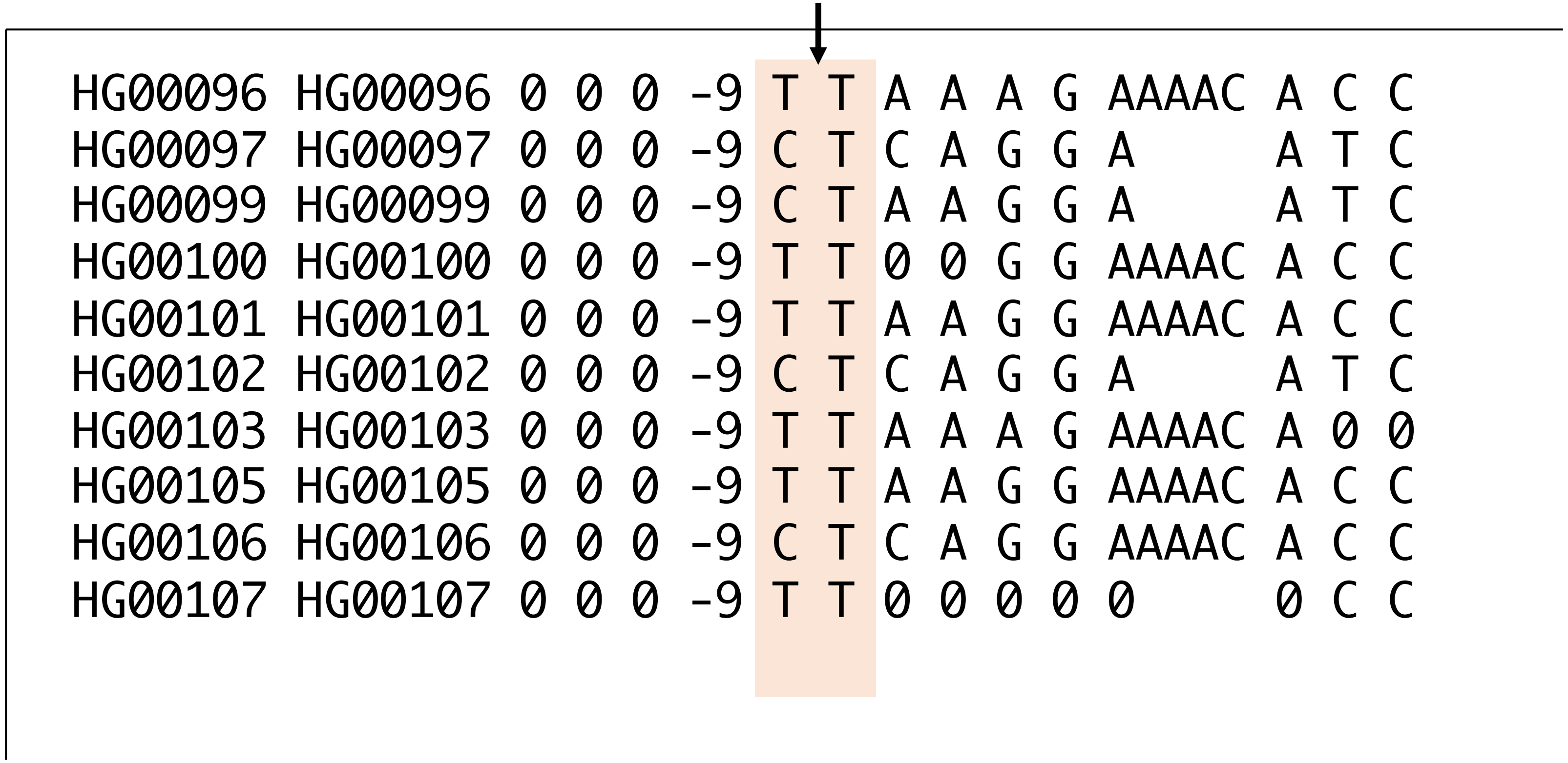


```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

All of the rest of the columns in the file give genotype data

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

Each pair of columns contains genotypes for a single variant; here's the first variant



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

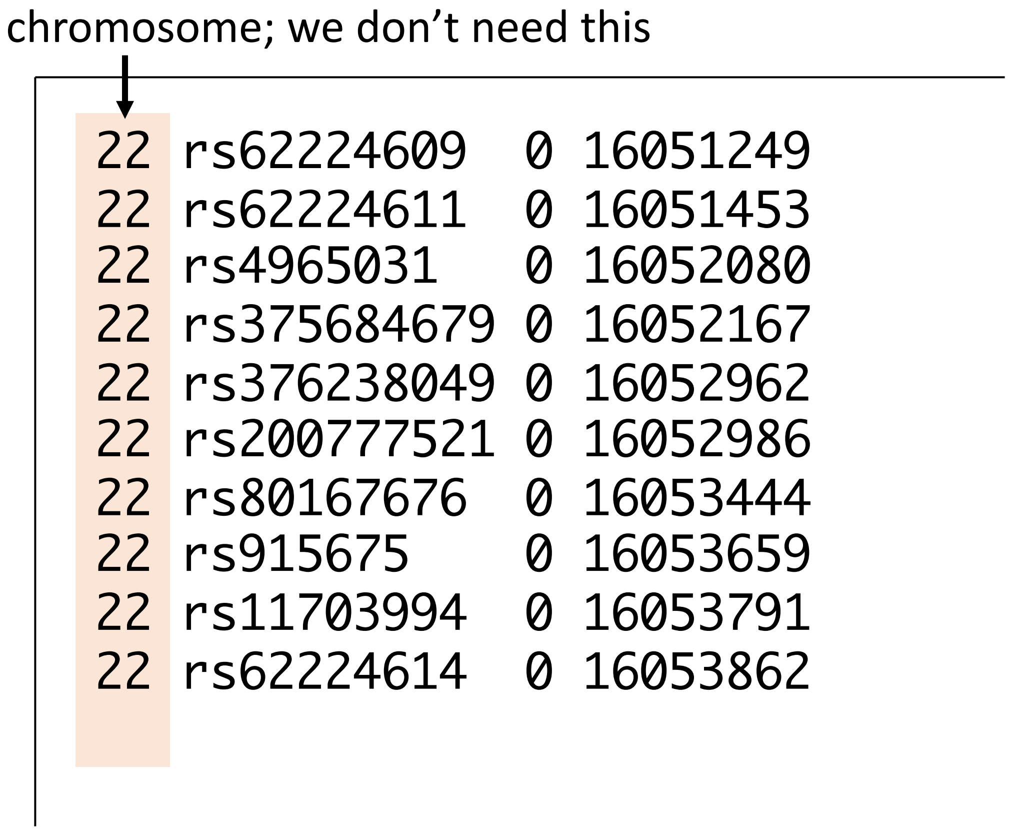
And the second

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

And the third, and so on

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

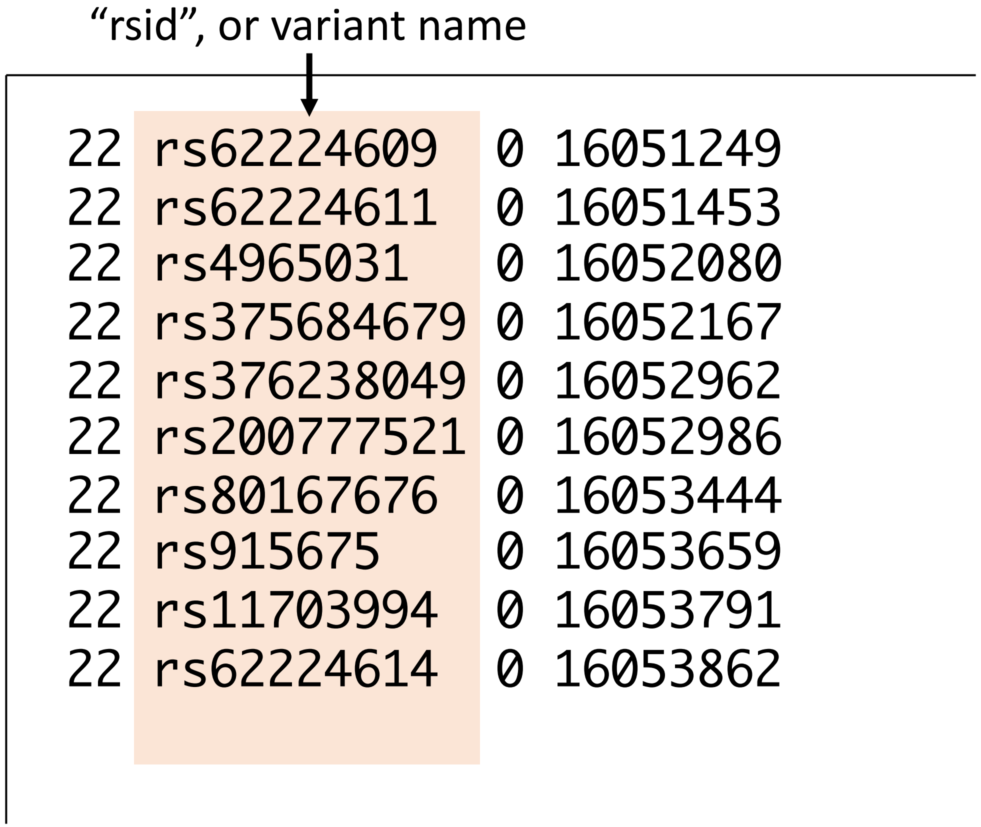
To get the names of the variants, you'll need to read in the map file. These are listed from top to bottom of this file in the same order they appear left to right in the ped file.



chromosome; we don't need this

```
22 rs62224609  0 16051249
22 rs62224611  0 16051453
22 rs4965031   0 16052080
22 rs375684679 0 16052167
22 rs376238049 0 16052962
22 rs200777521 0 16052986
22 rs80167676  0 16053444
22 rs915675    0 16053659
22 rs11703994  0 16053791
22 rs62224614  0 16053862
```

To get the names of the variants, you’ll need to read in the map file. These are listed from top to bottom of this file in the same order they appear left to right in the ped file.



“rsid”, or variant name

```
22 rs62224609  0 16051249
22 rs62224611  0 16051453
22 rs4965031   0 16052080
22 rs375684679 0 16052167
22 rs376238049 0 16052962
22 rs200777521 0 16052986
22 rs80167676  0 16053444
22 rs915675    0 16053659
22 rs11703994  0 16053791
22 rs62224614  0 16053862
```

To get the names of the variants, you'll need to read in the map file. These are listed from top to bottom of this file in the same order they appear left to right in the ped file.

more stuff we won't use (position in cM--missing here, and in basepairs)

```
22 rs62224609  0 16051249
22 rs62224611  0 16051453
22 rs4965031   0 16052080
22 rs375684679 0 16052167
22 rs376238049 0 16052962
22 rs200777521 0 16052986
22 rs80167676  0 16053444
22 rs915675    0 16053659
22 rs11703994  0 16053791
22 rs62224614  0 16053862
```

Let’s go back to the ped file.

So now we know from the map file that this is rs62224609

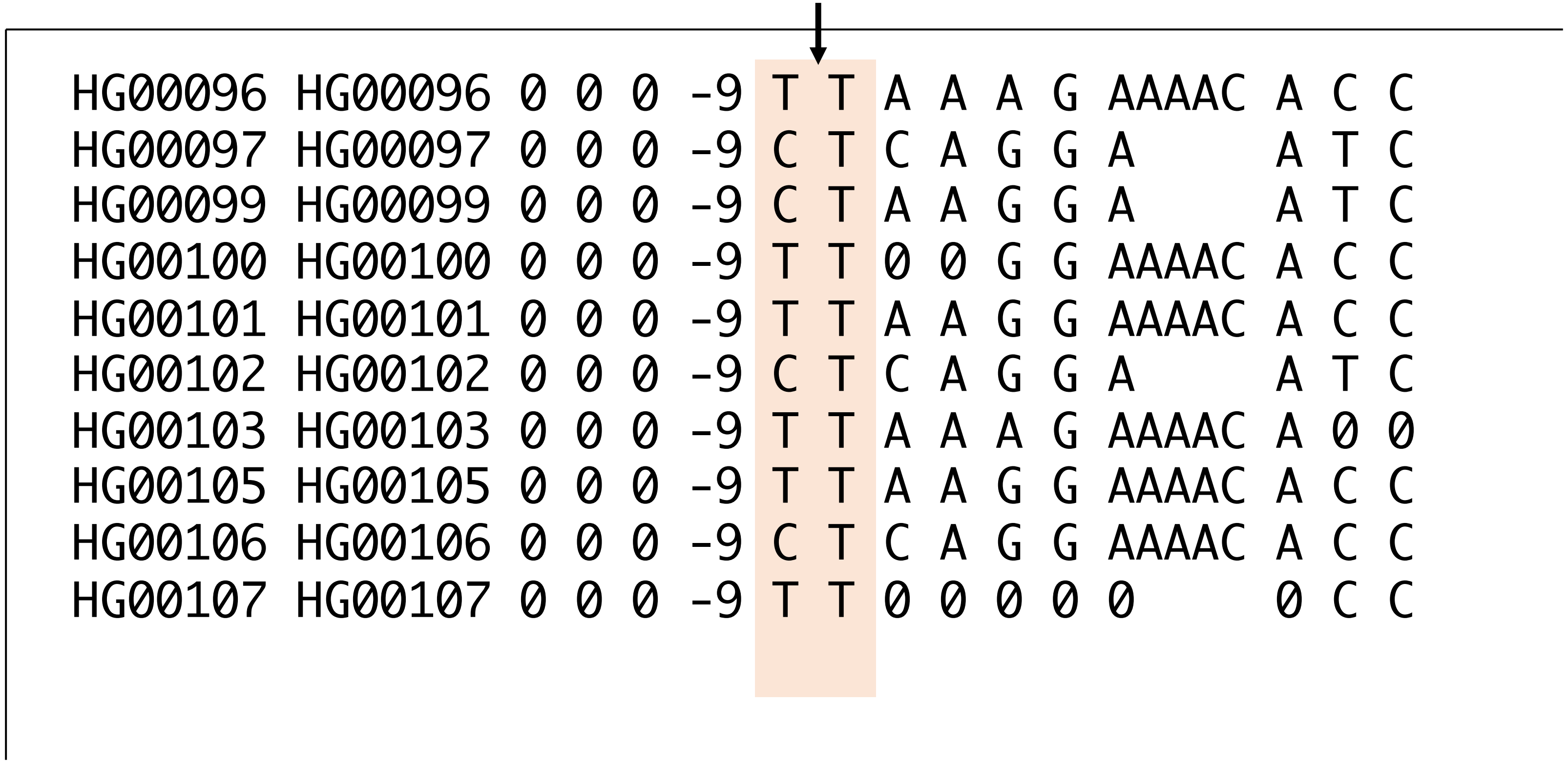
```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

And this is rs62224611

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

Each variant has two possible alleles.

For rs62224609, “T” and “C” are the possible alleles.



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

Each variant has two possible alleles.

Here’s an “indel”, or insertion-deletion. It has two alleles as well, one is just longer: “AAAAC” and “A”.

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

We can also have missing genotypes.

All pairs of “0 0” are missing values. Here are the missing values we can currently see:

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

You’ll want to calculate these statistics:

- Individual missingness
- Variant missingness
- Minor Allele frequency
- Bonus: Hardy-Weinberg equilibrium deviation test p-value

For individual missingness, you’ll need to tally the missing genotypes, or “0 0” pairs in the row for each individual. Remember, the first six columns are not genotypes, so don’t count any 0s in those.

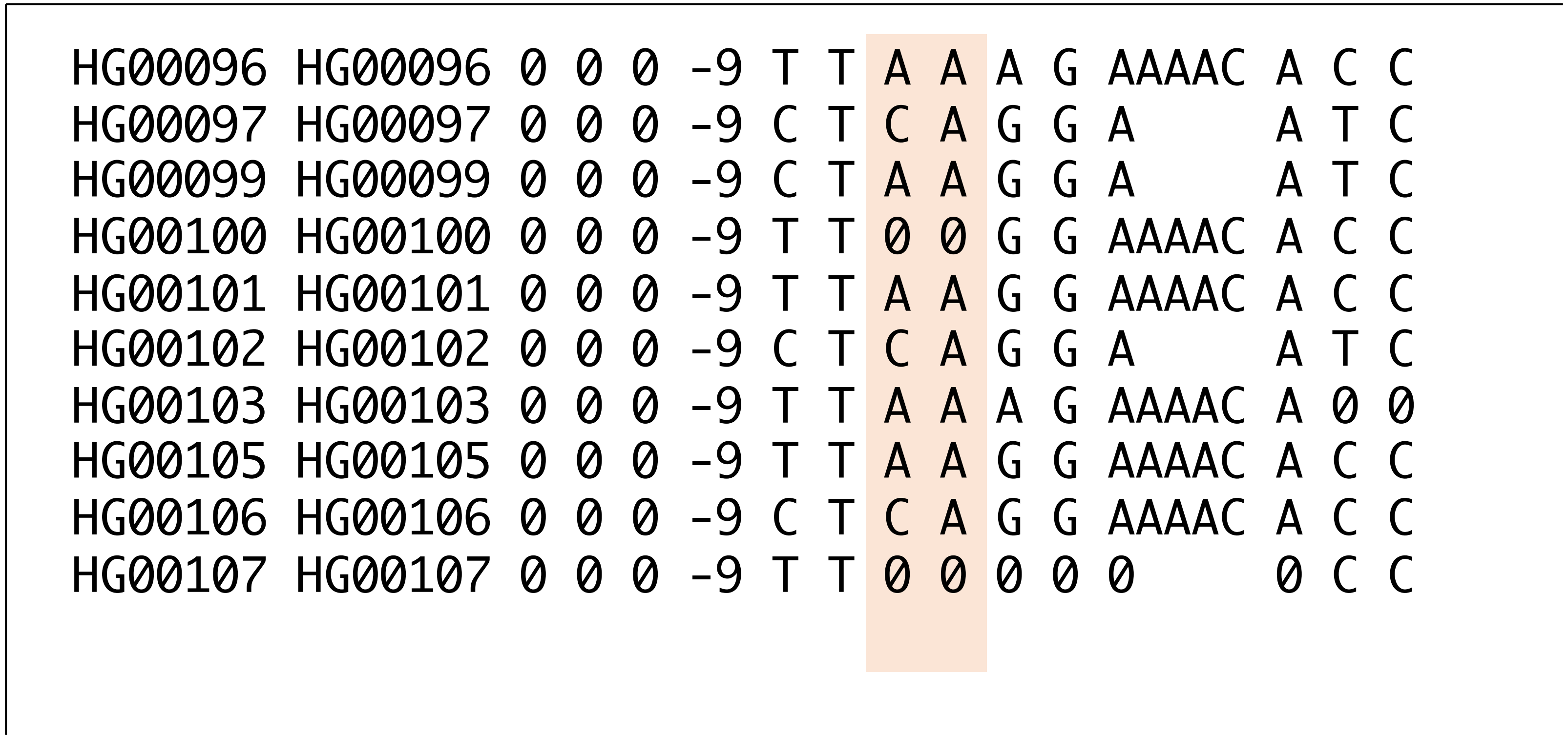
$$missingness = \frac{n_{missing}}{n_{total}}$$

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

E.g.: $missingness = \frac{3}{5} = 0.6$

For variant missingness, you’ll need to tally the missing genotypes, or “0 0” pairs in the pair of columns for each variant.

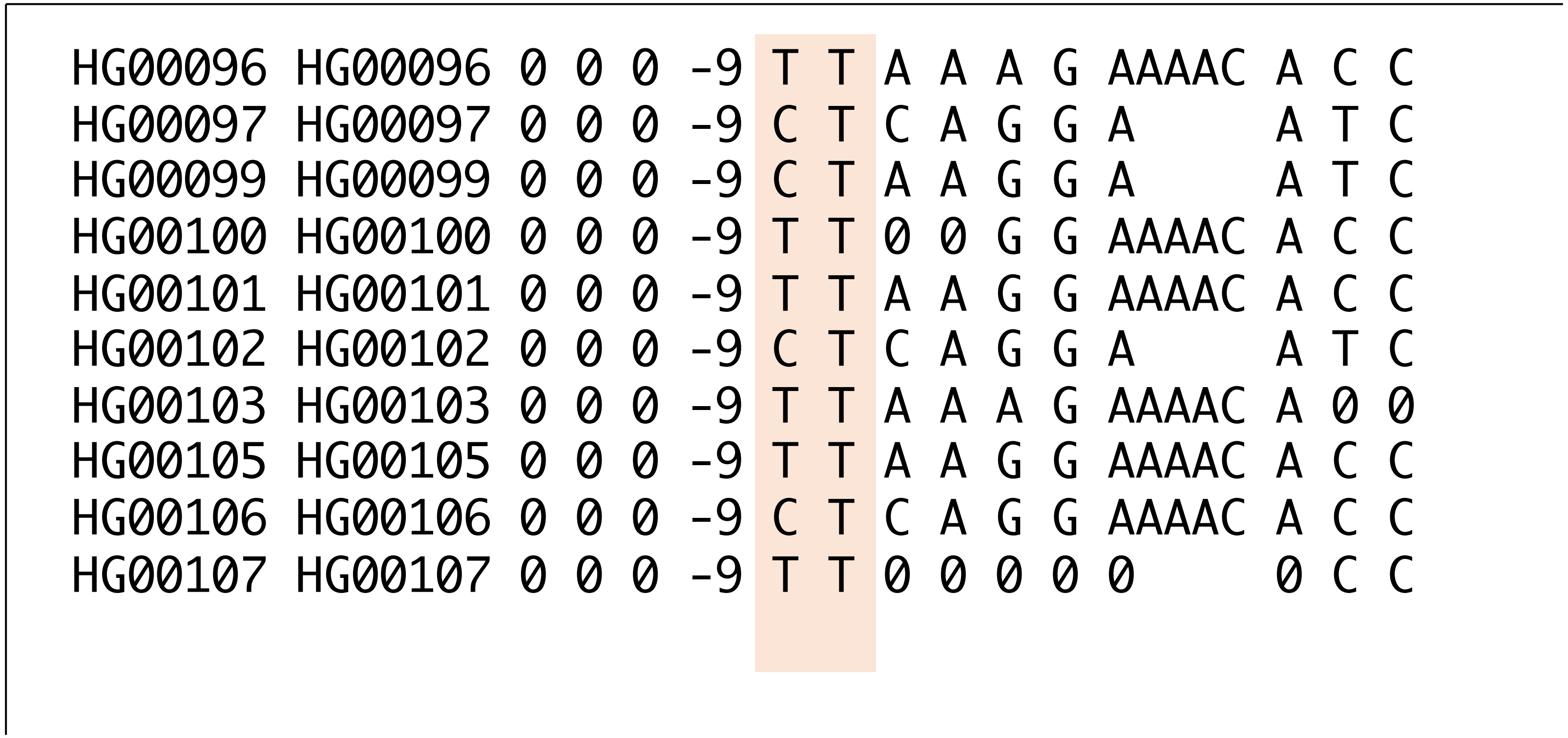
$$missingness = \frac{n_{missing}}{n_{total}}$$



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

E.g.: $missingness = \frac{2}{10} = 0.2$

For the Hardy-Weinberg equilibrium deviation test, you'll start by counting the number of each possible genotype observed for the variant. These are your "observed" values.



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

E.g.: TT: 6, CT: 4, CC: 0

Next, you'll use the observed values to calculate observed allele frequencies. I'll use fake larger counts here to demonstrate.

Observed counts: TT: 66, CT: 24, CC: 10

Sum the counts to get the total number of observed genotypes: $n = n_{TT} + n_{CT} + n_{CC} = 66 + 24 + 10 = 100$

Calculate the observed allele frequency for one allele, here, T: $AF_T = \frac{2n_{TT} + n_{CT}}{2n} = \frac{2(66) + 24}{2(100)} = 0.78$

The the observed allele frequency for the other allele, here, C: $AF_C = 1 - AF_T = 1 - 0.78 = 0.22$

Next, you'll use the observed values to calculate observed allele frequencies. I'll use fake larger counts here to demonstrate.

Observed counts: TT: 66, CT: 24, CC: 10

Sum the counts to get the total number of observed genotypes: $n = n_{TT} + n_{CT} + n_{CC} = 66 + 24 + 10 = 100$

Calculate the observed allele frequency for one allele, here, T: $AF_T = \dfrac{2n_{TT} + n_{CT}}{2n} = \dfrac{2(66) + 24}{2(100)} = 0.78$

The the observed allele frequency for the other allele, here, C: $AF_C = 1 - AF_T = 1 - 0.78 = 0.22$

And use the observed allele frequencies to calculate expected counts:

Expected count for TT: $n_{exp_{TT}} = n\left(AF_T{}^2\right) = 100\left((0.78)^2\right) = 60.84$

Expected count for CT: $n_{exp_{CT}} = n(2 \cdot AF_T \cdot AF_C) = 100(2 \cdot 0.78 \cdot 0.22) = 34.32$

Expected count for CC: $n_{exp_{CC}} = n\left(AF_C{}^2\right) = 100\left((0.22)^2\right) = 4.84$

Now use the observed and expected counts to calculate the Chi-square statistic

Observed counts: TT: 66, CT: 24, CC: 10

Expected counts: TT: 60.84, CT: 34.32, CC: 4.84

$$\chi^2 = \sum \frac{n_{obs} - n_{exp}}{n_{exp}}$$

$$\chi^2 = \frac{n_{obs_{TT}} - n_{exp_{TT}}}{n_{exp_{TT}}} + \frac{n_{obs_{CT}} - n_{exp_{CT}}}{n_{exp_{CT}}} + \frac{n_{obs_{CC}} - n_{exp_{CC}}}{n_{exp_{CC}}}$$

$$\chi^2 = \frac{66 - 60.84}{60.84} + \frac{24 - 34.32}{34.32} + \frac{10 - 4.84}{4.84} = 0.852$$

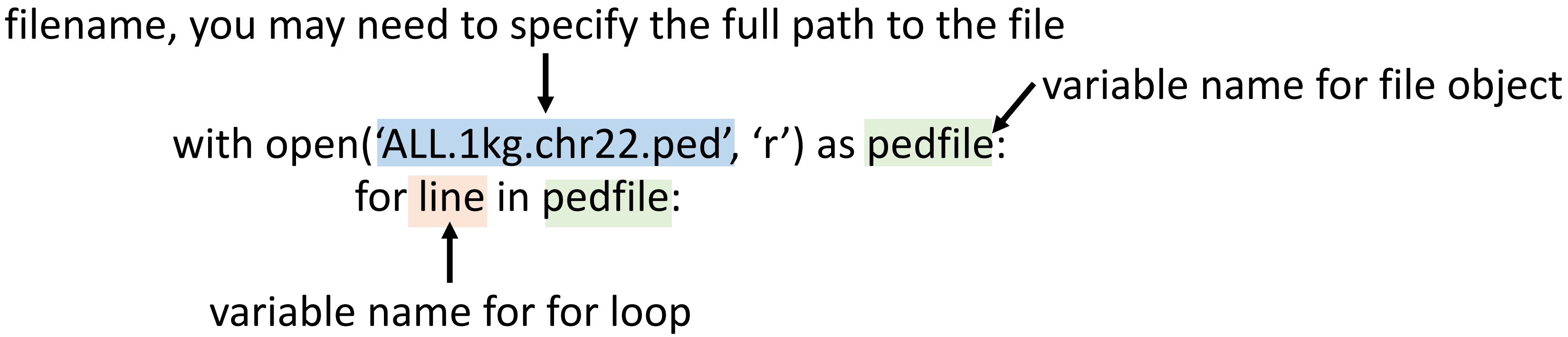
Use the Chi-square survival function, chi2.sf, in scipy.stats to get the p-value (df will always = 1):

```
from scipy.stats import chi2
sf(x=0.852, df=1)
```

When reading in large files in Python, it’s usually most efficient to read them line-by-line. The simplest syntax uses with:

```
with open(‘ALL.1kg.chr22.ped’, ‘r’) as pedfile:
        for line in pedfile:
```

When reading in large files in Python, it’s usually most efficient to read them line-by-line. The simplest syntax uses with:



filename, you may need to specify the full path to the file

variable name for file object

```
with open(‘ALL.1kg.chr22.ped’, ‘r’) as pedfile:
        for line in pedfile:
```

variable name for for loop

Calculating individual missingness will only require the data in one line at a time.

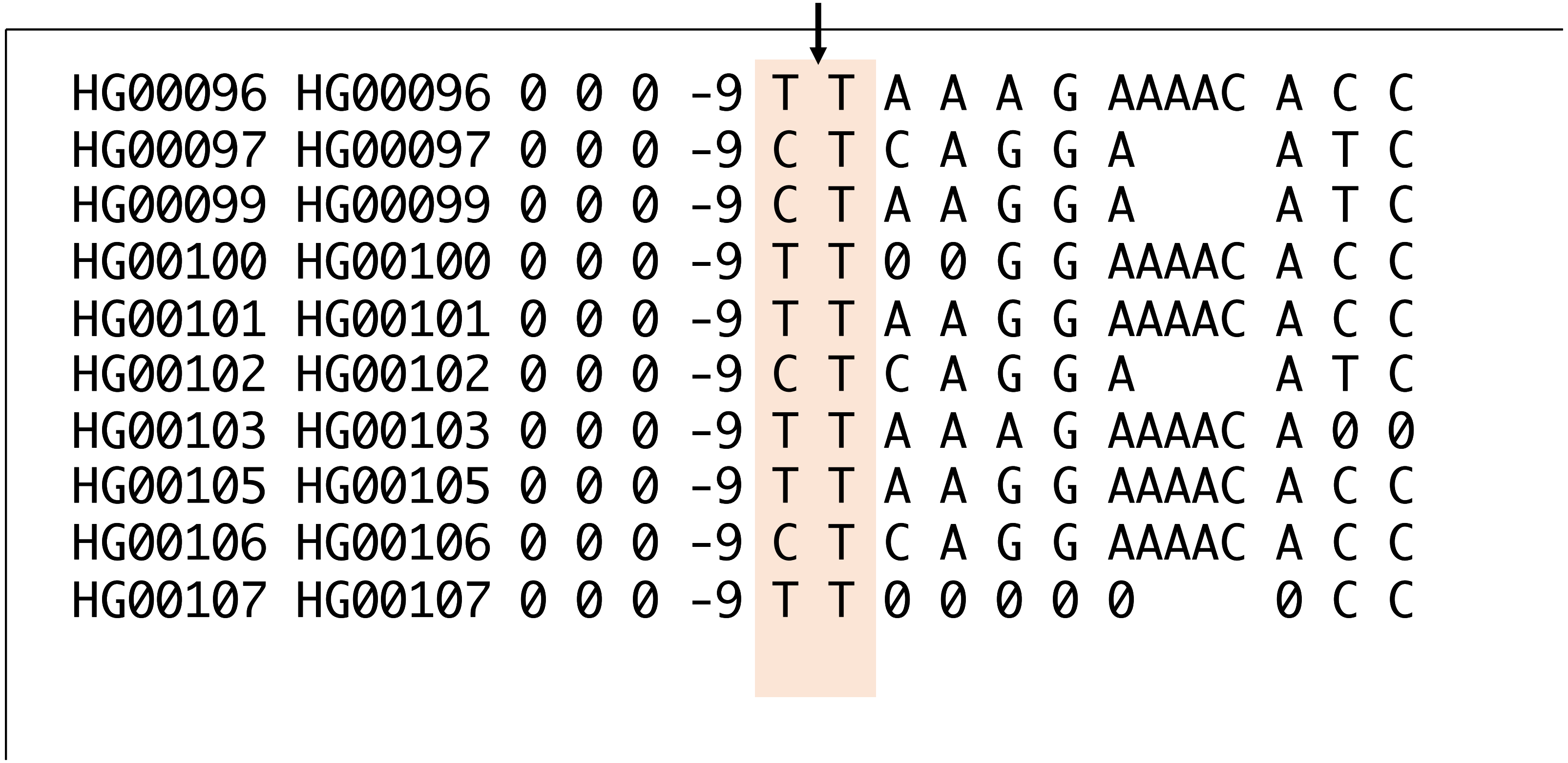
```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

You’ll need to loop through by 2; think about ways you can control that with range()

For variant missingness and HWE, you’ll have to figure out a way to aggregate the data from each pair of columns.

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

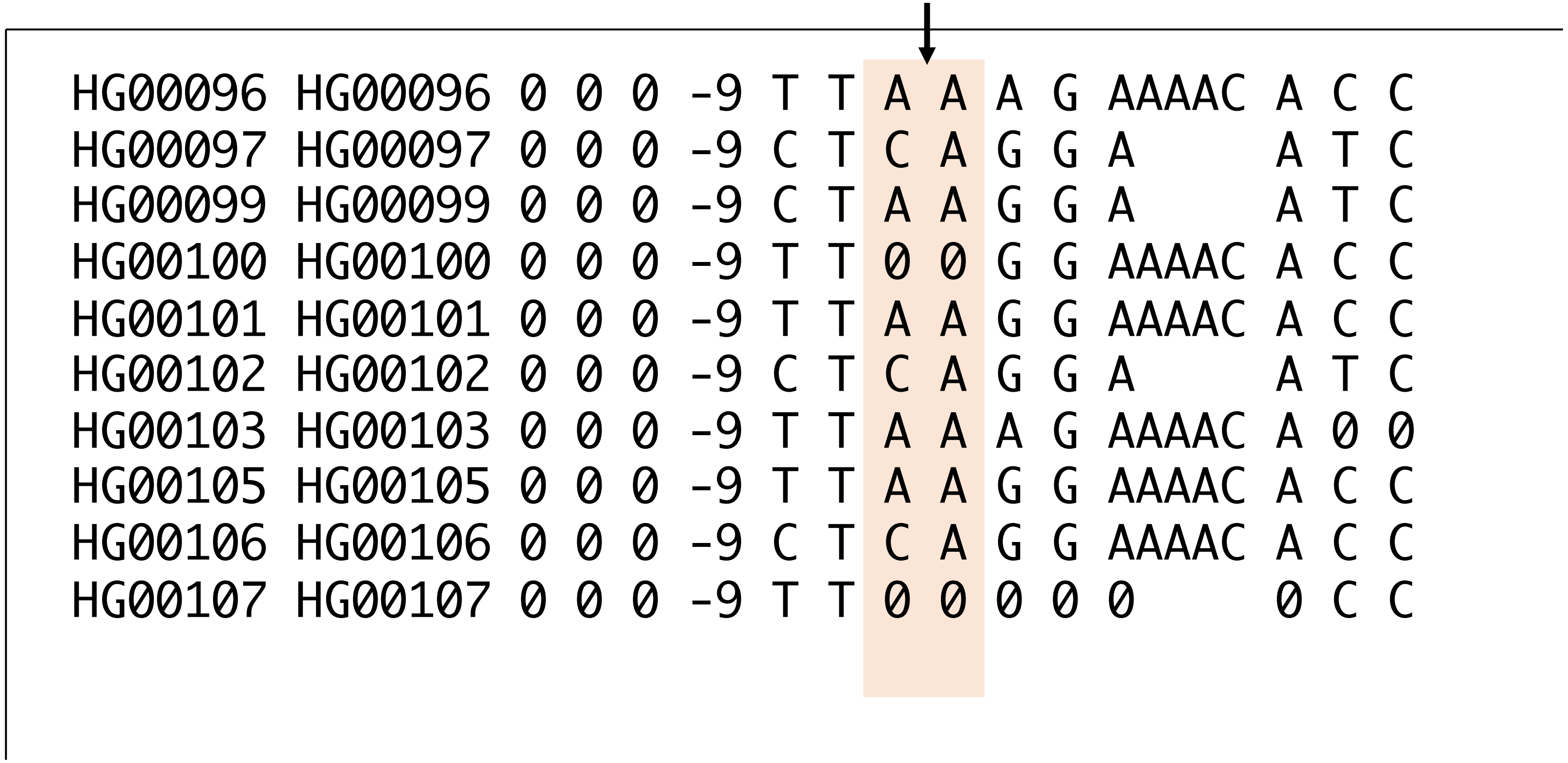
One way you could do that would be to create a dictionary that stores the genotypes for each variant. The key could be an index corresponding to the variant order in the pedfile, or it could be the variant ID.



So this would be 1 or rs62224609

```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

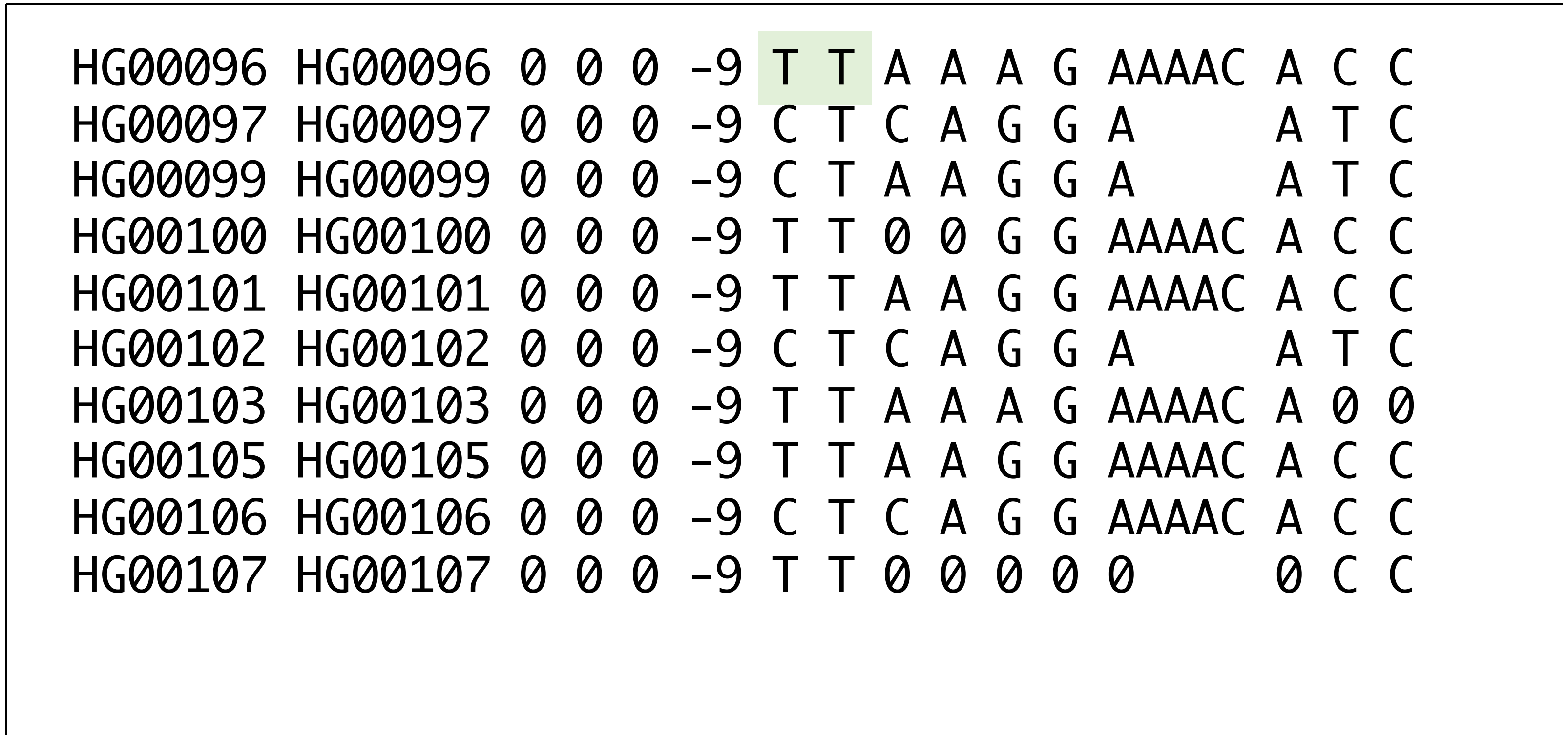
This would be 2 or rs62224611



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

As you process the genotypes in each line, you could add them to the dictionary for that variant. So for line one, you would add this genotype to the dictionary under key 1, so that the key-value pair looks something like this:

{1: [‘T T’]}



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

You could also have the genotype as a list, like {1: [[‘T’,’T’]]}

You’d then continue down the line, adding key-value pairs for each variant:

{1: [‘T T’]; 2: [‘A A’]}

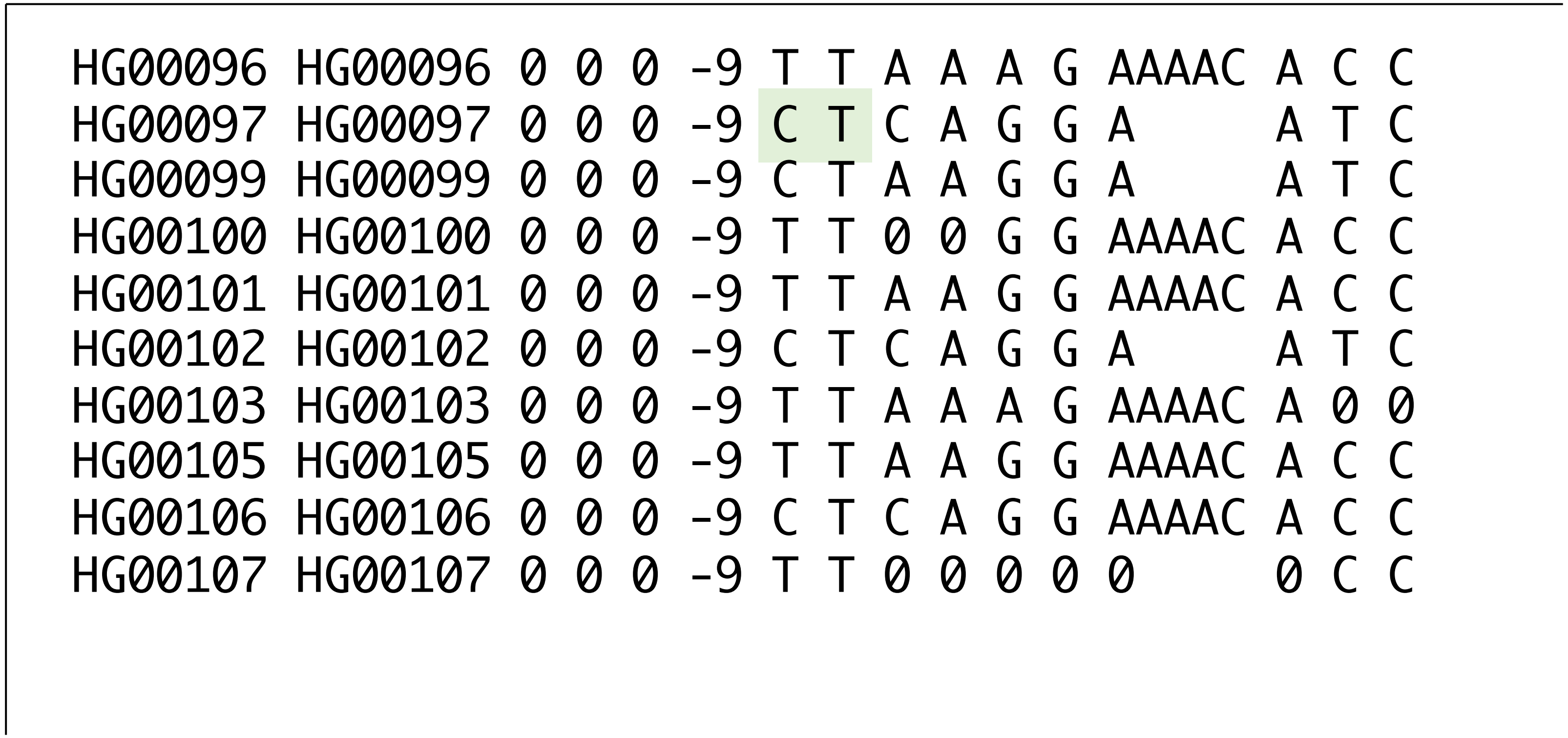
```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

Until you end up with:

{1: [‘T T’]; 2: [‘A A’]; 3: [‘A G’]; 4: [‘AAAAC A’], 5: [‘C C’]}

On the next line, you'd then append to that list for each variant (or you could initialize a list with enough elements for each individual). After adding the first genotype from this line, you would have:

{1: ['T T', 'C T']; 2: ['A A']; 3: ['A G']; 4: ['AAAAC A'], 5: ['C C']}



```
HG00096 HG00096 0 0 0 -9 T T A A A G AAAAC A C C
HG00097 HG00097 0 0 0 -9 C T C A G G A     A T C
HG00099 HG00099 0 0 0 -9 C T A A G G A     A T C
HG00100 HG00100 0 0 0 -9 T T 0 0 G G AAAAC A C C
HG00101 HG00101 0 0 0 -9 T T A A G G AAAAC A C C
HG00102 HG00102 0 0 0 -9 C T C A G G A     A T C
HG00103 HG00103 0 0 0 -9 T T A A A G AAAAC A 0 0
HG00105 HG00105 0 0 0 -9 T T A A G G AAAAC A C C
HG00106 HG00106 0 0 0 -9 C T C A G G AAAAC A C C
HG00107 HG00107 0 0 0 -9 T T 0 0 0 0 0     0 C C
```

And so on, until you've processed the whole file.