Introduction to Magma

We define a binary code as a four-dimensional subspace of the vector space K^7, where K is the binary field GF(2) with the following generator matrix. (The lines starting with a > show Magma commands)

```
> K := FiniteField(2); (alternatively: K :=GF(2);)
> C := LinearCode<K, 7 |
>    [1, 0, 0, 0, 1, 1, 1], [0, 1, 0, 0,  1, 1, 0],
>    [0, 0, 1, 0, 1, 0, 1], [0, 0, 0, 1, 0, 1, 1]>;
```

Now if you execute the line

```
>C;
```

you will get some information about C. To find the parity check matrix of C do this:

```
> H :=ParityCheckMatrix(C);
> H;
```

Remember that the usual definition of the parity check matrix is the transpose of our book's definition.
Now let's do some decoding using coset leaders. Conveniently, Magma has a lot of procedures (functions) already defined for us.

First we set L to be the set of coset leaders of C and f to be the map which maps the syndrome of a vector in V to its coset leader in L.

```
> L, f := CosetLeaders(C);
> L;
```

Since C has dimension 4, the degree of the information space I of C is 4. We set i to be an "information vector" of length 4 in I, and then encode i using C by setting w to be the product of i by the generator matrix of C.

```
> I := InformationSpace(C);
> I;
> i := I ! [1, 0, 1, 1];
> w := i * GeneratorMatrix(C);
> w;
```

The notation I ! [1,0,1,1] means the vector [1,0,1,1] is considered to be an element of the space I.
Now we set r to be the same as w but with an error in the 7-th coordinate (so r is the "received vector").

```
> r := w;
> r[7] := 0;
> r;
```

Finally we let s be the syndrome of r with respect to C, apply f to s to get the coset leader l, and subtract l from r to get the corrected vector v. Finding the coordinates of v with respect to the basis of C (the rows of the generator matrix of C) gives the original information vector.

```
> s := Syndrome(r, C);
> s;
> l := f(s);
> l;
> v := r - l;
> v;
> res := I ! Coordinates(C, v);
> res;
```