Exercise 1

Write a command line program in C++ with this syntax:

```
elias [c|d] <filein> <fileout>
```

The first parameter can be either "c" or "d", for compression or decompression.

When the "c" option is specified, the program opens the specified input file as text (the name is provided as the second command line parameter) and reads signed base 10 integers separated by whitespace. The program must then map each value to the range $[1, +\infty]$ using the following correspondence $(0, -1, 1, -2, 2, -3, 3, -4, 4, ...)$ to $(1, 2, 3, 4, 5, 6, 7, 8, 9, ...)$, i.e. negative numbers are mapped to even values, while non negative ones are mapped to odd numbers. The mapped numbers are then encoded with Elias γ code:

Value	Binary code	Code length
1	1	1
2	010	3
3	011	3
4	00100	5
5	00101	5
6	00110	5
7	00111	5
8	0001000	7
9	0001001	7
…	…	…

The output file is created in binary mode, with the name passed as the third parameter, and each number is saved with the number of bits specified by the encoding. The last byte is padded with 0, so that it cannot be read as a valid number.

When the "d" option is specified, the data is decoded from binary to text. The program must create a new file, with the name passed as the third parameter, with the same numbers saved in decimal text format, each followed by a new line.