# LZ4

LZ4 is a lossless data compression algorithm that is focused on compression and decompression speed. It belongs to the LZ77 family of byte-oriented compression schemes. The format of the compressed stream is defined in the two files `lz4_format_description.txt` and `lz4_explained.pdf`. A Windows executable `lz4_v1.3.3.exe` is also provided which allows you to compress and decompress files (it's an old version, but with an easier header, with respect to more recent versions).

With a bit of reverse engineering it was possible to understand that this program creates a header consisting of 4 bytes of magic identifier (in hex 03 21 4C 18), an unsigned 32-bit integer in little endian that indicates the length of the uncompressed file, a constant value (in hex 00 00 00 4D), then blocks of LZ4 compressed data.

Every block is made up of a 32-bit unsigned integer in little endian indicating the length of the block (`BlockLength`), followed by the bytes of the compressed stream. Notice that every block is composed of a sequence of Literal, MatchCopy, Literal, MatchCopy, ..., Literal. So, at the end of the block you will find a Literal only without a MatchCopy. After reading `BlockLength` bytes you will need to start again with a new block, so another BlockLength, then Literal, MatchCopy, ...
Each compressed block should provide at most 4 MB of uncompressed data.

Write a C++ command line program with the following syntax:

```
1. lz4decomp <input file> <output file>
```

The program reads an input file compressed with `lz4_v1.3.3.exe` and writes an output file with the decompressed data. If the number of command line arguments is wrong, the input file cannot be opened, the output file cannot be created, or the header doesn't start with the correct magic number, the program exits with return code 1, otherwise, after decompressing the file, it exits with return code 0. Nothing must be sent to standard output or standard error.