CLASSICAL AND QUANTUM COMPUTING
EXERCISE I

Intel hex record files are printable files consisting of any number of Intel hex records. Each record is represented by exactly one line of the format

```
:CCAAAARRDD...DZZ
```

The leading colon indicates that the line is an Intel hex record. The characters following the colon are hexadecimal (base 16) digits, i.e. `0` to `9` and `A` to `F`. The first to digits represented by `CC` determine the number of data bytes (represented by two hexadecimal digits each). The next four digits, `AAAA`, is an address. The two digits `RR` indicate the record type:

- `00` for a data record
- `01` for an end of file record
- `02` for an extended segment address record
- `03` for a start segment address record
- `04` for an extended linear address record
- `05` for a start linear address record.

This is followed by the data bytes. Lastly two hexadecimal digits `ZZ` give the checksum, which is the two's complement of the sum of all the previous bytes (modulo 256).

For example, the record

```
:020000021000EC
```

has 2 data bytes, and is a data record, with address 0. The checksum is

```
EC=14*16+12=236
```

The sum of the preceding bytes is

$$2 + 0 + 0 + 2 + 16 + 0 = 20$$

in decimal. In binary 20 is represented as

```
00010100
```

The 8 bit one's complement of this bitstring is

```
11101011
```

Thus the 8 bit two's complement is

```
11101100
```

This gives the result

$$2^7 + 2^6 + 2^5 + 2^3 + 2^2 = 128 + 64 + 32 + 8 + 4 = 236.$$

Write a program to read an Intel hex record file which uses the checksum of each record to determine if the record is correct. Use the program with the following file.

```
:100000008316FF30850000308600831202300920FD
:0C0010008600820741344234433444 34FB
:00000001FF
```