

# Computer Systems and -architecture

## Data Representation

1 Ba INF 2018-2019

Stephen Pauwels  
stephen.pauwels@uantwerpen.be

### Time Schedule

Exercises are made individually. Put all your files in a tgz archive, as explained on the course's website, and submit your solution to the exercises on Blackboard. **Write down all intermediate results on how you obtained the results.**

- Deadline: **November 8, 23u55**

### Exercises

1. Convert these positive numbers to base 10.
  - (a)  $(100101011)_2$
  - (b)  $(666E)_{16}$
  - (c)  $(1100110100111)_2$
  - (d)  $(123)_8$
  - (e)  $(666)_7$
2. Convert to base 10.
  - (a)  $(10011110)_2$  (2's complement)
  - (b)  $(11111)_2$  (2's complement)
  - (c)  $(0.2132)_4$
  - (d)  $(0.123)_{15}$
3. Convert to base 2.
  - (a)  $(2018)_{10}$
  - (b)  $(666)_8$
  - (c)  $(1A4D)_{16}$
  - (d)  $(9.87)_{10}$
  - (e)  $(42)_{16}$
4. Convert to base 2. Represent the negative numbers with 8 bits in *signed magnitude*, *one's complement*, *two's complement* and *excess 128*.
  - (a)  $(-100)_{10}$

- (b)  $(-68)_{10}$   
 (c)  $(-128)_{10}$   
 (d)  $(-2C)_{16}$
5. For the following single-precision IEEE 754 bit patterns, show the numerical value as a base 2 significand with an exponent (e.g.  $+1.11 \cdot 2^5$ ).
- (a) 0 10011000 001110110000000000000000  
 (b) 1 00111101 101100000000000000000000  
 (c) 1 11111111 000000000000000000000000  
 (d) 0 00000000 000010111000000000000000  
 (e) 1 00010100 111001100000000000000000  
 (f) 0 10101011 011010000000000000000000  
 (g) 0 11111111 11010100010001010100010
6. Represent these numbers in the *IEEE-754 (single precision)* format.
- (a)  $(8985.5)_{10}$   
 (b)  $(2018)_{10}$   
 (c) NaN  
 (d)  $(-42.666)_{10}$   
 (e)  $+\infty$   
 (f)  $+0$   
 (g)  $(1.110101 * 2^{-133})_2$  (denormalized)  
 (h)  $(333.0)_{10}$
7. Suppose we are using a 15 bit floating point, in a normalized, base 8 floating point format, with a sign bit, followed by a 5-bit exponent with a certain bias, followed by three base 8 digits.
- (a) Determine the bias we have to use for the exponent, assuming we do not want to change the range of exponents we would have reached when using a 5-bit 2's complement exponent.  
 (b) Represent the number  $(-142)_{10}$  in our new format (with the bias from the previous question) as a binary string.  
 (c) What is the largest possible error that can be made using this representation?  
 (d) What is the smallest gap using this representation?
8. Write a Python program that, using the module `files`, does the following:
- (a) Read the given file `input.txt` using the correct encoding.  
 (b) Write the contents you just read back to file using the UTF-16 encoding scheme.  
 (c) Convert all characters to their appropriate code points.  
 (d) Convert the code points to their correct html code, make sure you can display a new line in a correct manner.

The module `files` has the following functions:

- `read_file(filename, encoding)`: this function takes a filename and reads it according to the given encoding and returns the resulting string.
- `write_file(filename, contents, encoding)`: this function writes the string `contents` to the file `filename` using the given encoding.
- `write_html_file(filename, contents)`: writes the string `contents` to a html file.

The module has the following encodings:

- ASCII
- UTF\_8
- UTF\_16