Computer Systems and -architecture

Data Representation

1 Ba INF 2016-2017

Bart Meyers
bart.meyers@uantwerpen.be

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 10, 23u55**

Exercises

1. Convert these positive numbers to base 10.
   (a) $(10110111100)_2$
   (b) $(3A6E)_{16}$
   (c) $(111001)_2$
   (d) $(164)_8$
2. Convert to base 10.
   (a) $(11101011)_2$ (2's complement)
   (b) $(11111)_2$ (2's complement)
   (c) $(.213)_4$
3. Convert to base 2.
   (a) $(666)_{10}$
   (b) $(123)_8$
   (c) $(14AD)_{16}$
   (d) $(3.25)_{10}$
   (e) $(123)_{16}$
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) $(-123)_{10}$
   (b) $(-128)_{10}$
   (c) $(-68)_{10}$

(d) $(-7)_{\mathbf{16}}$

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 10001110 00111011000000000000000`

   (b) `1 10000000 10100000000000000000000`

   (c) `0 11111111 00000000000000000000000`

   (d) `0 00000000 00101110000000000000000`

   (e) `0 11111111 11010000010001000100000`

   (f) `1 00000111 11100110000000000000000`

   (g) `0 00001011 01101000000000000000000`

6. Represent these numbers in the *IEEE-754 (single precision)* format.

   (a) $(8082.5)_{10}$

   (b) $(2016)_{10}$

   (c) NaN

   (d) $(1.28)_{10}$

   (e) $+\infty$

   (f) $+0$

   (g) $(1.110101 * 2^{-135})_2$ (denormalized)

   (h) 666.0

7. Suppose we are using a 14 bit floating point, in a normalized, base 8 floating point format, with a sign bit, followed by a 4-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 4-bit 2's complement exponent.

   (b) Represent the number -157 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 functions 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`