Image Processing I – Exercises Week #2

Notes:

Try to solve the following questions (except for Exercise 2a/b) without the help of a computer. Enter your name, your student ID and your answers in this document and submit it as a Microsoft Word doc or pdf file to the following e-mail address: image.processing.jena@gmail.com.

Deadline for the submission is: Tuesday, November 17th, 2020, 8.00 a.m. (CET).

Name: .....

Student-ID: .....

Exercise 1: Conversion of decimal, binary, octal and hexadecimal numbers

a) Convert the following decimal numbers into the binary, octal and hexadecimal system:

Decimal	Binary	Octal	Hexadecimal
10	0000 1010	12	A
20	0001 0100	24	14
30	0001 1110	36	1E
40	0010 1000	50	28
127	0111 1111	177	7F
256	0001 0000 0000	400	100
1984	0111 1100 0000	3700	7C0
2020	0111 1110 0100	3744	7E4

b) Convert the following (unsigned) binary numbers into the decimal, octal and hexadecimal system:

Binary	Decimal	Octal	Hexadecimal
0000 0010	2	2	2
0000 0100	4	4	4
0000 1100	12	14	C
0001 0000	16	20	10
0001 1000	24	30	18
0011 1000	56	70	38
0111 1111	127	177	7F
1000 0000	128	200	80
1010 1010	170	252	AA
1111 1111	255	377	FF

c) Convert the following **signed** binary numbers into the decimal system assuming that negative numbers are represented by the two's complement

Binary (2's compl)	Decimal
0000 0010	2
0000 0100	4
1000 0000	-128
1010 1010	-86
1111 1111	-1

d) Fill the gaps in the following text by converting the binary and decimal numbers into the hexadecimal system:

Hi DAD (3501),

that's so BAD (1011 1010 1101). I am so tired, I am almost

DEAD (57005). I need to go to BED (1011 1110 1101) right now.

e) Most people can only count to 10 on their fingers. You, however, can now do better. If you regard each finger as one binary bit, with finger extended as 1 and finger touching the palm as 0, how high can you count using both hands?

All then fingers extended 1111111111 → 2^10 – 1 = 1023

Exercise 2: Data classes

a) Which is the lowest and highest value that can be stored in a Python float variable? How can you determine these values?

Python floats are double precision floating point numbers. The maximum value is (see slide 50):

OFL = (1-2^-(p+1)) * 2^(emax + 1)

With p=52, emax = 1023 (see slide 49) we have: OFL = 1.7976931348623157e+308

Python code:

```
>>> import sys
>>> print(sys.float_info.max)
```

b) Which is the smallest positive value that can be stored in a variable of type of float? How can you determine this value?

UFL = 2^emin

With emin = -1022 we have: UFL = 2.2250738585072014e-308

Python code:

```
>>> import sys
>>> print(sys.float_info.min)
```

Exercise 3: Arithmetic operations

a) Perform the following additions on (unsigned) binary numbers.

```
        00011101
    +   01101111
-------------------
Carry   111111
  Sum  10001100
```

```
        01000001
    +   11111111
-------------------
Carry   1111111
  Sum  101000000
```

b) Perform the following additions on signed binary numbers. (Note that in the two's complement notation the most significant bit represents the number -128. Negative numbers in the two's complement notation can be added like positive numbers and yield the correct result, when the overflow is ignored.)

```
        00011101          01000001
    +   01101111      +   11111111
-------------------  -------------------
Carry   11111110  Carry   11111110
  Sum   10001100    Sum  1 01000000
```

c) Perform the following additions on hexadecimal numbers.

```
        1234              FFFF
    +   ABCD          +   1111
-------------------  -------------------
Carry   0110      Carry  11110
  Sum   BE01        Sum  11110
```

d) A binary number can be subtracted by adding its two's complement, which is obtained by inverting all bits and adding 1. For example, using 1 byte (ie. 8 bits) the decimal number 6 is represented by 0000 0110. To find the signed binary number representing the decimal value -6, we first invert the bits: 1111 1001 (one's complement) and then add 1: 111 1010 (two's complement).

Now, subtract B = 0101 0101 and C = 0111 0101 from A = 0111 0000 by adding the two's complement of B and C to A.

1's complements: B → 1010 1010, C → 1000 1010

2's complements: B → 1010 1011, C → 1000 1011

```
  A     01110000          A     01110000
- B     10101011        - C     10001011
-------------------  -------------------
Carry   110000000 Carry   00000000
  Sum  100011011    Sum   11111011
```