2018 EXAMINATIONS

Lancaster University

Part I (First Year)

SCHOOL OF COMPUTING AND COMMUNICATIONS

SCC.150 Digital Systems (1 hour & 30 Minutes)

- **Answer any <u>THREE</u> out of the four questions.**
- **Use a <u>separate</u> answer book for <u>each</u> question.**

Question 1

1.a Describe the problem we end up with when encoding extremely large or extremely small fixed point numbers (i.e. fractions) and how it is solved on a von Neumann machine. **[3 marks]**

1.b i. Convert the decimal number 186_{10} to its corresponding unsigned 8-bit binary number.

[1 mark]

ii. Convert the unsigned 8-bit binary number 11110110_2 to its corresponding decimal number.

[1 mark]

iii. Convert the decimal number 94_{10} to its corresponding hexadecimal number. **[1 mark]**

iv. Convert the following decimal numbers to 8-bit 2's complement: 35_{10}, 80_{10} . Then add them, giving the answer also in 8-bit 2's complement.

[2 marks]

v. Convert the following decimal numbers to 8-bit 2's complement: -40_{10} , -55_{10} . Then add them, giving the answer also in 8-bit 2's complement.

[2 marks]

1.c i. Specify the decimal number -128_2 in an 8-bit binary sign and magnitude representation.

[2 marks]

ii. Specify the hexadecimal BA_{16} in an 8-bit binary sign and magnitude representation.

[2 marks]

iii. Convert the decimal number $10/32_2$ to the 32-bit IEEE 754 floating point and express your answer in hexadecimal. (Reminder: the 32 bits are used as follows: Bit 1: sign of mantissa, bits 2-9: 8-bits of exponent in excess 127, bits 10-32: 23 bits for magnitude of mantissa.)

[6 marks]

Question 1 continues on next page…

1.d i. Describe what are the status flags, what is their purpose and where they are located in a Central Processing Unit (CPU) of a von Neumann machine.

[2 marks]

ii. Name the three basic status flags that are used by the Arithmetic Logic Unit (ALU). **[3 marks]**

[Total 25 Marks]

Question 2

2.a What does the term RISC stand for? What is the characteristic of a RISC architecture? What does MIPS stand for?

[3 marks]

2.b Write a MIPS instruction that initializes the register $t0 to the value 20.

[2 marks]

2.c Convert the instruction below to its MIPS machine language representation (32 bits). Show the value for each of the 4 fields. Give the full 32 bit representation of the instruction in hexadecimal format.

addi $t1, $t2, 16

ADD immediate: *Add register **rs** and the sign-extended immediate value **imm** and store the result in register **rt**.*

addi rt, rs, imm

0x8	rs	rt	imm
6	5	5	16

Name	Register number
$zero	0
$v0–$v1	2–3
$a0–$a3	4–7
$t0–$t7	8–15
$s0–$s7	16–23
$t8–$t9	24–25
$gp	28
$sp	29
$fp	30
$ra	31

[8 marks]

Question 2 continues on the next page…

2.d The address 0x12345678 should be loaded into register $t0. Write the MIPS assembler code for this task without using MIPS pseudo-instructions. Please comment the code you provide

[5 marks]

2.e Write a MIPS assembler program which carries out the following calculation:

$$x = 2*a - b$$

The result x should be available in register $s1 at program completion. Please explain your register mapping. You can assume that a and b are present in registers (no loading from memory is required).

[7 marks]

[Total 25 Marks]

Question 3

3.a State the output of the program below. Explain the differences between the operators &, &&, | and ||.

```
#include <stdio.h>
char a = 0x10;
char b = 0x20;
int main() {
    if (a & b) {
        printf("a &  b is true\n");
    }
    if (a && b) {
        printf("a && b is true\n");
    }
    if (a | b) {
        printf("a |  b is true\n");
    }
    if (a || b) {
        printf("a || b is true\n");
    }
}
```

[6 marks]

3.b Implement a function strdup which creates a copy of a null-terminated string. The strdup function should accept as an argument a pointer to a null-terminated string and it must create a copy of that string, by allocating sufficient memory and copying the content of the string to the newly allocated memory. The function should return a pointer to the new string.

```
char *strdup(char *s) {
<<YOUR CODE>>
}
```

[7 marks]

3.c What is a device file? Describe the two device file types found in Unix systems and discuss their differences.

[6 marks]

Question 3 continues on the next page…

3.d Describe the four primary capabilities of the preprocessing language. The content of a header and a source file are presented below. State the output of the program.

```
File example.c:
#include <example.h>
int main() {
#if __PRINT__
    printf("Test print
      line\n");
#endif
    printf("Hello world!");
}
```

```
File example.h:
#include <stdio.h>

#define __PRINT__ 1
```

[6 marks]

[Total 25 Marks]

Question 4

4.a Consider the following Boolean expression:

F = ABCD + A'BCD + ABC'D + A'BC'D + ABC'D' + A'BC'D' + A'B'C'D' + AB'C'D' + AB'CD'

i. Minimise it using a Karnaugh map and demonstrate clearly the Karnaugh map table, the identified groups extracted from the table as well as the final, simplified version of the Boolean expression using that approach.

[4 marks]

ii. Finally, apply De Morgan's law on the simplified Boolean expression from question 4.a i) and provide the new resulted expression.

[3 marks]

4.b Memory is organized in regions called Text, Data, Stack and Heap. The program below defines the variables counter and i. In which memory region does the content of each variable reside? In which region is memory allocated with **`malloc`**?

```
#include <stdio.h>
#include <stdlib.h>

int counter = 0;

int inc () {
    counter++;
}

int main() {
   int i;
   char *tmp;
   for (i = 0; i < 10; i++)
      inc();
   tmp = malloc(counter);
   strcpy(tmp, "hello!");
   return 0;
}
```

[8 marks]

Question 4 continues on the next page…

4.c Write a MIPS assembler program which has equivalent functionality to the following C code. At program termination, the result of variable a should be available in register $s0.

```
int a=1;
int i=0;
for (; i<5; i++){
    a=a<<1;
}
```

[10 marks]

[Total 25 Marks]

---END OF PAPER---