Coláiste na Tríonóide, Baile Átha Cliath
Trinity College Dublin
Ollscoil Átha Cliath | The University of Dublin

Faculty of Engineering, Mathematics and Science

School of Computer Science & Statistics

Integrated Computer Science **Trinity Term 2016**
Computer Science & Business
Computer Science & Language
Year 1 Annual Examination

CS1021 – Introduction to Computing I

10 May 2016 **RDS** **14:00 – 16:00**

Adam Taylor

Instructions to Candidates

Answer any **TWO out of THREE questions**.
All questions are marked out of 25.
Answer each question in a new answer booklet.
Where you are asked to write an assembly language program, you must provide suitable comments to explain your program.

Permitted Materials

An **ARM Instruction Set and Addressing Mode Summary** booklet is available on request.
Non-programmable calculators are permitted for this examination. You must indicate the make and model of your calculator on the front of your first answer booklet.

1. a) Convert the following numbers to their 16 bit hexadecimal 2's compliment representation.

   I) $+31612_{10}$ [1 mark]

   II) -555_{10} [1 mark]

   III) $+111000111100_2$ [1 mark]

   b) In a base 13 representation of numbers $[0, \longrightarrow 9, X, Y, Z]$ represent $[0_{10}, \longrightarrow 9_{10}, 10_{10}, 11_{10}, 12_{10}]$. Using this representation convert the following:

   I) $+YZ3_{13}$ to decimal [2 marks]

   II) $+236_{10}$ to base 13 [2 marks]

   c) Consider the following sequence of ARM Assembly Language instructions. For each highlighted instruction, **give the final value in the destination register** and **state whether each of the N (Negative), Z (Zero), C (Carry) and V (oVerflow) flags is set or clear** after the execution of the instruction. Answers without a brief supporting explanation or calculation will receive zero marks. [6 marks]

```
1        LDR     R7, =0xEEEEEEEF
2        LDR     R8, =0x22222221
3        ADDS    R6, R7, R8 ; Condition Code Flags? Value in R6?

4
5        LDR     R1, =0xB000FF00
6        LDR     R2, =0xC000EE00
7        ADDS    R0, R1, R2 ; Condition Code Flags? Value in R0?
8
9        LDR     R4, =0x33000000
10       LDR     R5, =0xB1000001
11       SUBS    R3, R4, R5 ; Condition Code Flags? Value in R3?
12
13       LDR     R10, =0x12300060
14       SUBS    R9, R10, R11 ; Condition Code Flags? Value in R9?
```

question 1 continued on next page ...

... *question 1 continued from previous page*

d) I) Translate the pseudo-code shown below into ARM Assembly Language. Assume that a, b, total are stored in R0, R1, R2 respectively. [8 marks]

```
total = 0
while(a > 0)
{
    a= a >> 1        ;shift right 1 place
    if(oneShiftedOut())
    {
        total=total+b
    }
    b = b << 1       ;shift left 1 place
}
```

II) What is the effect of this program? [2 marks]

III) Will it work correctly with signed numbers stored in 2's compliment? Explain your answer. [2 marks]

2. This question is about accessing memory and operating on ASCII data.

   Pilish is a type of constrained writing. In it, each word's number of letters corresponds to a digit of pi (π). The particular digit of π is determined by the word's location in a Pilish passage. For example, the first word in a valid Pilish sentence's number of letters corresponds to the first digit of π, therefore, it is three. The second word is one letter and so on. For example, the following is the start of a Pilish poem corresponding to 3.14159265358979.

   > But a time I spent wandering in bloomy night; Yon tower, tinkling chimewise, loftily opportune. — Joseph Shipley

   In Pilish words are separated by spaces and punctuation does not count. When a zero occurs in π, this corresponds to a word with ten digits.

   Design and write an ARM Assembly Language program that will check whether ASCII text stored in memory is valid Pilish or not. Valid Pilish is text that conforms to the rules without breaking them. The ASCII text is NULL terminated and stored in memory at the label PILISH and the address is stored in R1. Words are separated by exactly one space. The digits of π are stored as word sized values in memory at the label DIGITS and the address is stored in R2. Valid Pilish is indicated by storing a 1 in R0, invalid Pilish is indicated by storing a 0 in R0.

   Your answer must include the following:

   I) An explanation of your approach [7 marks]

   II) Your ARM Assembly Language program [18 marks]

3. This question is about accessing memory and bit manipulation.

   a) I) Provide ARM Assembly Language to set bit 14 of R6 leaving other bits unchanged. [2 marks]

   II) Provide ARM Assembly Language to flip bits 1, 3, 5 and 9 of R4 leaving other bits unchanged. [2 marks]

   b) Design and write an ARM Assembly Language program that will determine the minimum number of bits that a number stored in 96 bits requires. Consider the following example, the number in Data 1 is 8 bits long, but only 5 bits are required as there are 3 leading zeros (left most bit is most significant), so the answer in this case is 5.

0	0	0	1	1	1	0	1

Data 1: 8 bits of storage

Assume that the number to test is stored in memory at the label NUMBER with the address in R5. The test number is stored with the most significant byte at the lowest address and the least significant at the highest address. Store the answer in R3. Your program should work for any value stored in 96 bits. Your answer must include the following:

I) An explanation of your approach [7 marks]

II) Your ARM Assembly Language program [14 marks]