

George Theodoridis (S5951178)

INSTRUCTIONS

1. Submission is only via Themis for the practical exercises and via Brightspace for the theoretical exercises. Deadlines are strict.
 2. The exercises in this assignment add up to 100 points. To calculate your grade simply divide the number of points by 10.
 3. You must submit a pdf typeset in (La)TeX (no handwritten solutions) using **this** template.
 4. Seeking solutions from the internet, from any external resource, or from any other person is prohibited.
 5. Please note that the course lecturer reserves the right to ask the student submitting the assignment to explain the answers to any or all questions. If the student is unable to provide a satisfactory answer then that question may receive partial/no credit.
 6. Of course, university policies on plagiarism always apply. In particular, any suspected plagiarism will be reported to the Board of Examiners.
-

1. You are given the following LC3 ASM program! For this exercise, you have two tasks: **(10 points)**
 1. Write the symbol table (use an absolute address, not offsets)
 2. Manually assemble (aka convert to the internal LC3 representation) the instructions at A, B and D. Both binary and hex are fine.

```
.ORIG x3000
    AND R0, R0, #0
A   LD R1, E
    AND R2, R1, #1
    BRp C
B   ADD R1, R1, #-1
C   ADD R0, R0, R1
    ADD R1, R1, #-2
D   BRp C
    ST R0, F
    TRAP x25
E   .BLKW 1
F   .BLKW 1
    .END
```

Solution:

1.

```

x3000      AND    R0,R0,#0

x3001      A: LD    R1,E
x3002      AND    R2,R1,#1
x3003      BRp    C

x3004      B: ADD   R1,R1,#-1

x3005      C: ADD   R0,R0,R1
x3006      ADD    R1,R1,#-2

x3007      D: BRp    C
x3008      ST     R0,F
x3009      TRAP   x25

x300A      E: .BLKW 1

x300B      F: .BLKW 1

```

2.

```

0010 001 000001000
0101 010 001 1 00001
0000 001 000000001
0001 001 001 1 11111
0001 000 000 000 001
0001 001 001 1 11110
0000 001 111111101
0011 000 000 000000010
1111 0000 0010 0101
000000000000000000
000000000000000000

```

2. For this exercise we will assume that two new operations have been added to LC3: PUSH and POP. The instruction PUSH Rx pushes the value in Register x onto the stack. POP Rx removes a value from the stack and loads it into Rx. You are also given a list of the instructions which were executed, however some of the registers went missing. Fill out what the correct registers would be! **(10 points)**

BEFORE		AFTER	
R0	x0000	R0	x1111
R1	x1111	R1	x1111
R2	x2222	R2	x3333
R3	x3333	R3	x3333
R4	x4444	R4	x4444
R5	x5555	R5	x5555
R6	x6666	R6	x6666
R7	x7777	R7	x4444

Operations:

```

PUSH R4
PUSH (a)
POP (b)

```

```
PUSH (c)
POP R2
POP (d)
```

Solution:

```
PUSH R4
PUSH R1
POP R0
PUSH R3
POP R2
POP R7
```