# SEMESTER 1 SAMPLE EXAMINATION 2018/2019

**MODULE:**      CA296 - Compt. Archit. & Assembly Level Programming

**PROGRAMME(S):**
CASE          BSc in Computer Applications (Sft.Eng.)
ECSAO         Study Abroad (Engineering & Computing)

**YEAR OF STUDY:** 2,O

**EXAMINER(S):**

Donal Fitzpatrick                (Internal)            (Ext:8929)

**TIME ALLOWED:**   3 Hours

**INSTRUCTIONS:**    Answer all questions.
**Please use the files provided to answer the questions.**
**You are allowed to refer to materials which you have brought into the examination venue.**
**Please be sure to save your work to the H: drive.**

**PLEASE DO NOT TURN OVER THIS PAGE UNTIL YOU ARE INSTRUCTED TO DO SO.**
The use of programmable or text storing calculators is expressly forbidden.

*There are no additional requirements for this paper.*

# QUESTION 1                                                    [TOTAL MARKS: 20]

Please open the file named **q1.txt**. Here, you will find spaces where you may type the answers to the following questions. Be sure to save this file at regular intervals. Also, make sure that you are saving to the **h:** drive.

### Q 1(a)                                                            [2 Marks]
What is meant by the MNZ instruction? When would you use this instruction?

### Q 1(b)                                                            [2 Marks]
If a simple processor has an 8-bit data bus and a 10-bit address bus, what is the maximum amount of memory that the processor can access?

### Q 1(c)                                                            [2 Marks]
Explain, with an example, how a programmer might control the placement of their code in memory.

### Q 1(d)                                                            [2 Marks]
What is meant by the term **interrupt**? Give one example of why you might use an **interrupt**.

### Q 1(e)                                                            [2 Marks]
What is meant by the **SHR** instruction? What is the arithmetic effect of this instruction?

### Q 1(f)                                                            [2 Marks]
The SR contains the data for 5 flags. Name each, and explain, in not more than one sentence for each flag, what its purpose is.

### Q 1(g)                                                            [2 Marks]
When a **ret** instruction is used at the end of a procedure, what happens?

**Q 1(h)** [2 Marks]

Under what circumstances would you use each of the following:
- (I) AND
- (II) OR
- (III) XOR
- (IV) NOT

**Q 1(i)** [2 Marks]

Explain the purpose of the IP.

**Q 1(j)** [2 Marks]

What is meant by the term **Machine Code**? Give one reason why you, as a programmer, might use **Machine Code** instead of **Assembly**.

*[End of Question1]*

**QUESTION 2** *[TOTAL MARKS: 20]*

Please open the file named **q2.asm**. Use this file to answer the following questions. Make sure to save your work regularly. Make sure you are saving to the **h:** drive.

In the following question, you should be aware of the following:
1. The numeric keypad behaves like the main interrupt-driven keyboard.
2. It is accessed by INT04, and Port 08.

**Q 2(a)** [10 Marks]

Write a program which accepts a value 1-8 from the numeric keypad. The number should then light the corresponding bars on the right-hand seven-segment display. You should include appropriate error-handling to ensure that all other characters entered are ignored.

**Q 2(b)** [10 Marks]

Add to the code from the previous part to do the following:

1. Allow the user to enter a value 1-9 on the numeric keypad.
2. This value wil control the number of times the program repeats.
3. Now, cause the number which you entered in the first part of the question, to be repeatedly displayed on the right-hand seven-segment display. You should use the timer interrupt to ensure that it stays there for one second.

So for example: suppose you entered the number 4, then the number 5. This means that the number 4 is displayed 5 times, and that it stays on the seven-segment display for one second, before the display is cleared.

**[End of Question2]**

## QUESTION 3          [TOTAL MARKS: 20]

Please read the instructions in the following question carefully, as you must use two separate files to answer it. In all cases, make sure to save your files regularly. Make sure you are saving to the **h:** drive.

**Q 3(a)**                                                                                      **[12 Marks]**
Open the file named **q3a.**asm an use it to answer the following question. Write a program which accepts a lowercase character using the **simple keyboard**. Only lowercase characters should be accepted, no others. Once it has been correctly read, you should convert the character to uppercase, and display it on the top left of the VDU.

Marks will be awarded for good program design, use of procedures, and any other techniques you wish to incorporate.

**Q 3(b)**                                                                                      **[8 Marks]**

Open the file named **q3b.txt** an use it to answer the following question.

Explain what is meant by **the stack**. Explain how it is used, why it is used, and what features of Assembly Language may be utilised to interact with it.

**[End of Question3]**

**QUESTION 4**                                                    *[TOTAL MARKS: 20]*

Please read the instructions in the following question carefully, as you must use two separate files to answer it.  In all cases, make sure to save your files regularly.  Make sure you are saving to the **h:** drive.

**Q 4(a)**                                                                **[8 Marks]**
Open the file named **q4a.**asm and use it to answer the following question.  Write a procedure which accepts two numbers placed in AL and BL as parameters. The program should print the procedure should print the smaller of the two numbers to the top left of the VDU.

Write a short test program which verifies that the procedure works, using the numbers 0X05 and 0X06.

**Q 4(b)**                                                               **[12 Marks]**
Open the question named **q4b.**asm and use it to answer the following question. Write a procedure which determines the smallest number in a list of numbers stored in memory locations [A0] to [A9] inclusive.  Write a test program to fill that area of memory with numbers (use DB) and then to call the procedure. (Hint: The procedure from part(a) could be useful here, and you may copy the code and reuse it if you wish).

**[End of Question4]**

## QUESTION 5                                                    *[TOTAL MARKS: 20]*

Please read the instructions in the following question carefully, as you must use two separate files to answer it.  In all cases, make sure to save your files regularly.  Make sure you are saving to the **h:** drive.


## Q 5(a)                                                               [6 Marks]


Open the question named **q5a.txt** and use it to answer the following question.
Outline three techniques for passing parameters to a procedure in assembly. Write a brief note on the advantages and disadvantages of each.


## Q 5(b)                                                               [14 Marks]


Open the question named **q5b.**asm and use it to answer the following question.
Write a procedure to delay a program by a configurable number of seconds (you may
assume that the timer interrupt is configured to fire every second). The procedure should accept the number of seconds to delay as a parameter. You may decide on which mechanism you are using to pass the parameter.

[End of Question5]

## ASCII TABLE

Dec = Decimal (Base 10),  Hex = Hexadecimal (Base 16),  Oct = Octal (Base 8),  Char = Character

```
Dec Hex Oct Char  Binary           Dec Hex Oct Char  Binary
-----------------------------------------------------------------
 0  00  000 NUL   0000 0000         64  40  100 @     0100 0000
 1  01  001 SOH   0000 0001         65  41  101 A     0100 0001
 2  02  002 STX   0000 0010         66  42  102 B     0100 0010
 3  03  003 ETX   0000 0011         67  43  103 C     0100 0011
 4  04  004 EOT   0000 0100         68  44  104 D     0100 0100
 5  05  005 ENQ   0000 0101         69  45  105 E     0100 0101
 6  06  006 ACK   0000 0110         70  46  106 F     0100 0110
 7  07  007 BEL   0000 0111         71  47  107 G     0100 0111
 8  08  010 BS    0000 1000         72  48  110 H     0100 1000
 9  09  011 HT    0000 1001         73  49  111 I     0100 1001
10  0A  012 LF    0000 1010         74  4A  112 J     0100 1010
11  0B  013 VT    0000 1011         75  4B  113 K     0100 1011
12  0C  014 FF    0000 1100         76  4C  114 L     0100 1100
13  0D  015 CR    0000 1101         77  4D  115 M     0100 1101
14  0E  016 SO    0000 1110         78  4E  116 N     0100 1110
15  0F  017 SI    0000 1111         79  4F  117 O     0100 1111

16  10  020 DLE   0001 0000         80  50  120 P     0101 0000
17  11  021 DC1   0001 0001         81  51  121 Q     0101 0001
18  12  022 DC2   0001 0010         82  52  122 R     0101 0010
19  13  023 DC3   0001 0011         83  53  123 S     0101 0011
20  14  024 DC4   0001 0100         84  54  124 T     0101 0100
21  15  025 NAK   0001 0101         85  55  125 U     0101 0101
22  16  026 SYN   0001 0110         86  56  126 V     0101 0110
23  17  027 ETB   0001 0111         87  57  127 W     0101 0111
24  18  030 CAN   0001 1000         88  58  130 X     0101 1000
25  19  031 EM    0001 1001         89  59  131 Y     0101 1001
26  1A  032 SUB   0001 1010         90  5A  132 Z     0101 1010
27  1B  033 ESC   0001 1011         91  5B  133 [     0101 1011
28  1C  034 FS    0001 1100         92  5C  134 \     0101 1100
29  1D  035 GS    0001 1101         93  5D  135 ]     0101 1101
30  1E  036 RS    0001 1110         94  5E  136 ^     0101 1110
31  1F  037 US    0001 1111         95  5F  137 _     0101 1111
```

```
32  20  040  SPace 0010 0000      96   60  140  `     0110 0000
33  21  041  !     0010 0001      97   61  141  a     0110 0001
34  22  042  "     0010 0010      98   62  142  b     0110 0010
35  23  043  #     0010 0011      99   63  143  c     0110 0011
36  24  044  $     0010 0100      100  64  144  d     0110 0100
37  25  045  %     0010 0101      101  65  145  e     0110 0101
38  26  046  &     0010 0110      102  66  146  f     0110 0110
39  27  047  '     0010 0111      103  67  147  g     0110 0111
40  28  050  (     0010 1000      104  68  150  h     0110 1000
41  29  051  )     0010 1001      105  69  151  i     0110 1001
42  2A  052  *     0010 1010      106  6A  152  j     0110 1010
43  2B  053  +     0010 1011      107  6B  153  k     0110 1011
44  2C  054  ,     0010 1100      108  6C  154  l     0110 1100
45  2D  055  -     0010 1101      109  6D  155  m     0110 1101
46  2E  056  .     0010 1110      110  6E  156  n     0110 1110
47  2F  057  /     0010 1111      111  6F  157  o     0110 1111

48  30  060  0     0011 0000      112  70  160  p     0111 0000
49  31  061  1     0011 0001      113  71  161  q     0111 0001
50  32  062  2     0011 0010      114  72  162  r     0111 0010
51  33  063  3     0011 0011      115  73  163  s     0111 0011
52  34  064  4     0011 0100      116  74  164  t     0111 0100
53  35  065  5     0011 0101      117  75  165  u     0111 0101
54  36  066  6     0011 0110      118  76  166  v     0111 0110
55  37  067  7     0011 0111      119  77  167  w     0111 0111
56  38  070  8     0011 1000      120  78  170  x     0111 1000
57  39  071  9     0011 1001      121  79  171  y     0111 1001
58  3A  072  :     0011 1010      122  7A  172  z     0111 1010
59  3B  073  ;     0011 1011      123  7B  173  {     0111 1011
60  3C  074  <     0011 1100      124  7C  174  |     0111 1100
61  3D  075  =     0011 1101      125  7D  175  }     0111 1101
62  3E  076  >     0011 1110      126  7E  176  ~     0111 1110
63  3F  077  ?     0011 1111      127  7F  177  DEL   0111 1111
```

*[END OF EXAM]*