INTERNATIONAL ISLAMIC UNIVERSITY, ISLAMABAD
FACULTY OF BASIC and APPLIED SCIENCES
DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

Terminal Examinations, Spring 2020

Course Title: Computer Organization and Assembly Language	**Course Code: CS322**
Program: BS Computer Science	**Batch: BSCS F17 A & B**
Total Marks: (60 marks for BS /MA / M.Sc. (50 marks for MBA/ MS /Ph.D.)	**Date & Time: 15-Aug-2020 (04:00 pm)**
Credit Hours: 04	**Teacher Name: Mr. Asim Munir**

Q. No.	Marks Obtained
1	
2	
3	
Total for Semester Exam	
Mid-term Marks	
Total Marks out of 100	

Student Name: Muhammad Abdullah Kamran
Student Registration Number: 4037-FBAS/BSCS4/F18(A)
04 Credit Hours

Instructions for Students:

Before starting your open book examination, please read all the given below instructions carefully, and must follow these instructions carefully. You must affirm the honesty pledge given at the end:

1. Download the question paper titled as **"COAL Question Paper.pdf" (pdf file)** and answer-sheet titled as **"COAL_Answer-Booklet.docx" (MS Word document)** from the Google Classroom as per instructions of your teacher. You are required to write down the answers to each question in your own handwriting on neat white papers with any blue pen.
2. **Maximum time to download question paper, attempt and submit/ upload your answer sheets is 8 HOURS. As soon as you finish your paper** Upload your answer booklet on priority basis. soon. You can only upload your exam response **once.** You will be unable to re-upload an additional

or amended version. If you fail to submit it within the due time, your paper will be considered cancelled.

3. **How to submit(upload) your answer-booklet/paper**:

   After completing your answers, you need to:

   a. Mention/write your **Name** and **Registration Number**, **Page number** and **sign** on each page of your handwritten answer-sheet.
   b. Take pictures using mobile camera or Scan each page of your written answers /answer sheets via any scanning software (as guided in the video tutorial).
   c. Insert all pictures or scanned images of your answer sheets into the MS word file titled as **"COAL_Answer-Booklet.docx"** provided by the teacher in the Google Classroom.
   d. After inserting all the images, save the **"COAL_Answer-Booklet.docx"** file as a single PDF file **(Only PDF format is acceptable as your answer-booklet),** and upload it in the Google Forms (link of which is provided in the Google Classroom).
   e. Please make sure you upload the correct document as you will not be able to change this, once it has been submitted.
      (Please see the video tutorial regarding procedure to upload the examination responses, shared in the Google classroom).

4. The University views copying from one another's examination paper/ cheating, giving or receiving unpermitted aid, discussion/consultation, plagiarism, impersonation during an examination, as serious disciplinary offences that may fall under the category of Use of Unfair Means and will be dealt as per university rules for UMCC.

5. Before starting your examination, you must agree to and sign the following pledge by having a click on the Student's Affirmation check box (it is mandatory to Tick the Checkbox):

   "I hereby affirm that i) I shall solve this paper on my own and I shall not seek the help of any person(s) with any sort of aid (like telephonic/verbal help, attempted answers related to my examination etc.) while taking my paper, (ii) or will not provide assistance of any sort (verbal or written) to other fellow students. If I am found involved in i) cheating ii) impersonation, iii) or using plagiarized content in my writing, my case may be dealt as per university rules and procedures for using unfair means."

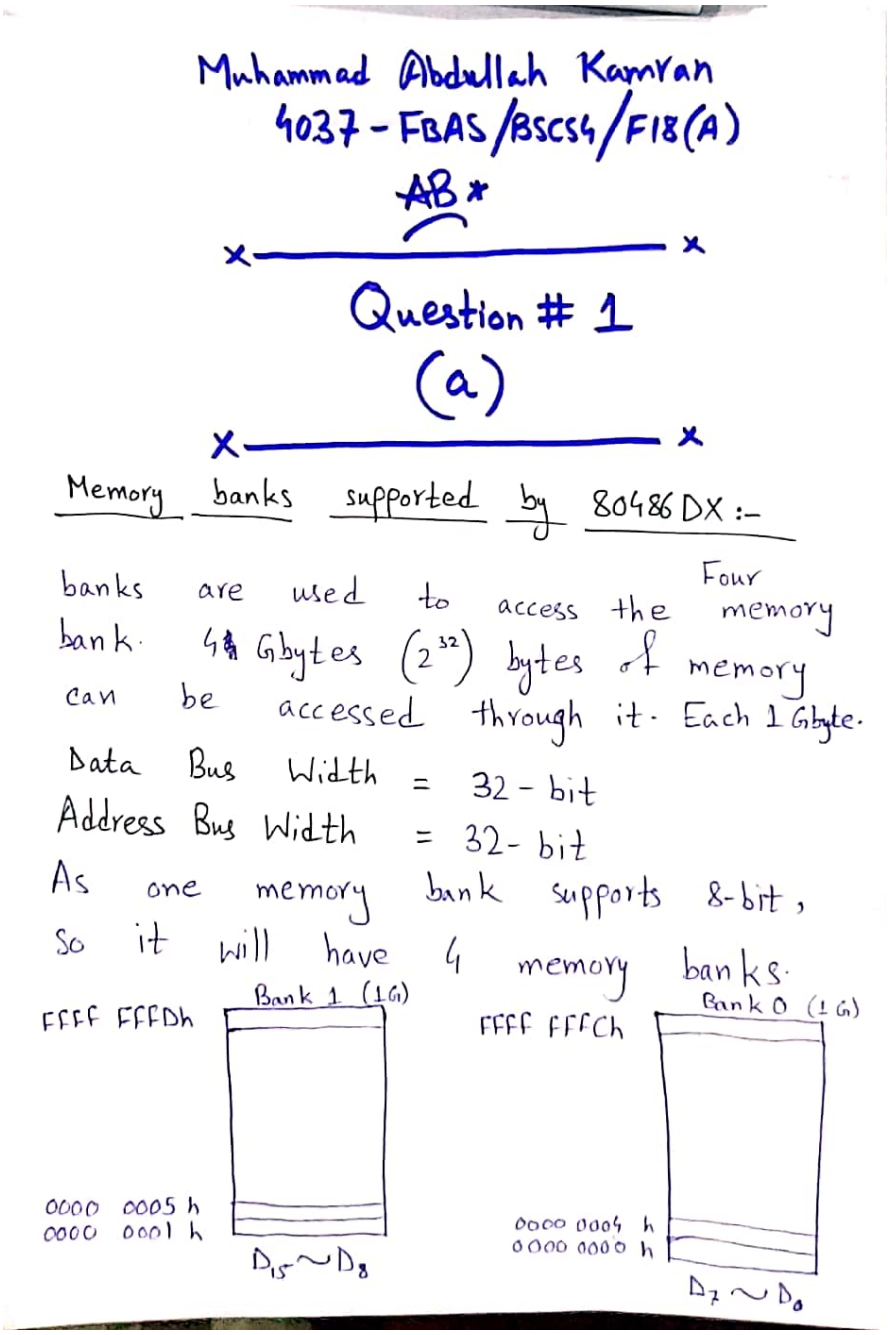
   Student's Affirmation: I affirm the given pledge (√)

[Start Inserting Images on Page No. 3]

Q1.

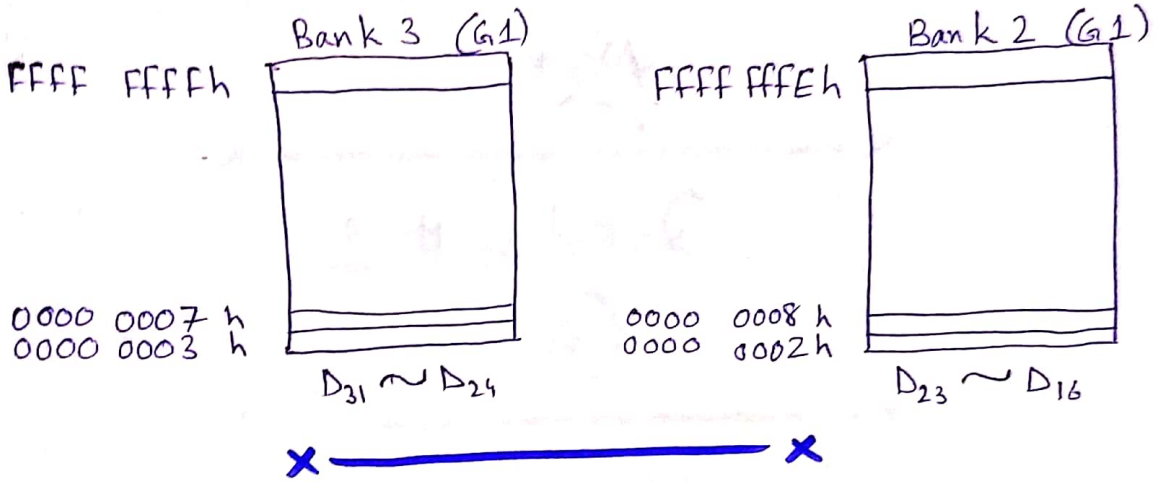
Insert Pictures of Answer Sheet Here

(A)



M. Abdullah Kamran
4037 - FBAS/BSCS/F18(A)

AB *



(B)

Muhammad Abdullah Kamran
4037-FBAS/BSCS4/F18(A)

AB

Question # 1
part (b)

Code segment starts at 000A C9ADh

Physical Address = 000A × 10h + C9ADh
= 0CA4Dh + C9ADh
= 0CA4Dh (Base Address)

Size = 0C70h
limit = 0CA4Dh + 0C70h
= 0D6BDh

Binary conversion for Base = 0CA4Dh

0000 0000 | 0000 0000 | 1100 1000 | 0010 1101
$B_{31} \sim B_{24}$ | $B_{23} \sim B_{16}$ | $B_{15} \sim B_8$ | $B_7 \sim B_0$

Binary conversion for limit = 0D6BD

0000 | 1101 0110 | 1011 1101
$l_{19} \sim l_{16}$ | $l_{15} \sim l_8$ | $l_7 \sim l_0$

Descriptor Table :-

7	0000 0000 ($B_{31} \sim B_{24}$)	G=0, D=1, 0, A/V=1, 0000 ($l_{19} \sim l_{16}$)	6
5	(Access Rights) 1 1 1 1 1 1 0	0000 0000 ($B_{23} \sim B_{16}$)	4
3	1100 1010 ($B_{15} \sim B_8$)	0010 1101 ($B_7 \sim B_0$)	2
1	1101 0110 ($l_{15} \sim l_8$)	1011 1101 ($l_7 \sim l_0$)	0

(C)

Muhammad Abdullah Kamran
4037-FBAS/BSCS 4/F-18(A)

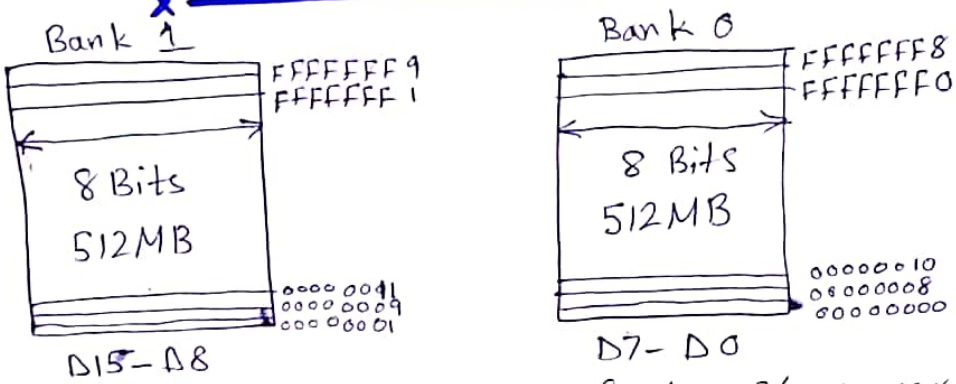
Question # 1

(c)

Memory banks enabled through Address lines of Intel's Pentium Processor :-

Muhammad Abdullah Kamran
4037-FBAS/BSCS4/F18(A)

AB*



Pentium processors was the first x86 processor with superscalar architechture. It also features a 64-Bit external data bus, which doubles the amount of information it is possible to read or write on each memory access. Pentium processor contains 2 full processor's combine into one with each of 32-bits. So, its overall data requirment was 64-bit. So, it has 64 data bit. It has 2 ALU's doing 32-bit operations.

Address lines Needed for decoding :-

- For pentium processors we have 8 locations and each have 8-bits.
- If we assume No. of address lines is (n) = 1 we can only address 2 locations (0 & 1)
- If n=2 we can address 4 locations (0, 1, 2, 3)
- No. of address locations 2^n.
- Here Address locations are 8, so 8 = 2^n
- So, n = log (8) to the base 2 = 3

Therefore, 3 address lines are needed for decoding.

(D)

M. Abdullah Kamran
4037-FBAS/BSCS4/F18(A)

AB

Question # 1

(d)

Enteries a selector can point :-

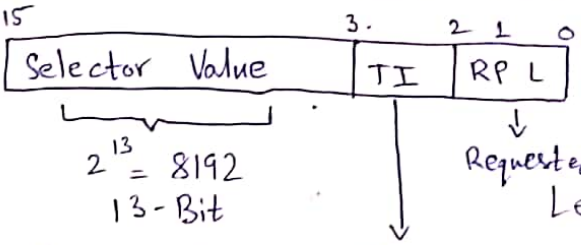
A selector can point out 8192 entries. As the maximum entries that a selector can point out are 8192 and selector selects those from descriptor table which is of 8 bytes, so,

$8 \times 8192 = 64\,KB$

Therefore, a descriptor table requires 64KB of Memory. Total Entries $= \frac{64k}{8} = 8k = 8192$

Detemining Size of Descriptor Table:-

Sdctor also tells us about tabbe indictor



$2^{13} = 8192$
13-Bit

Requested Privilege Level

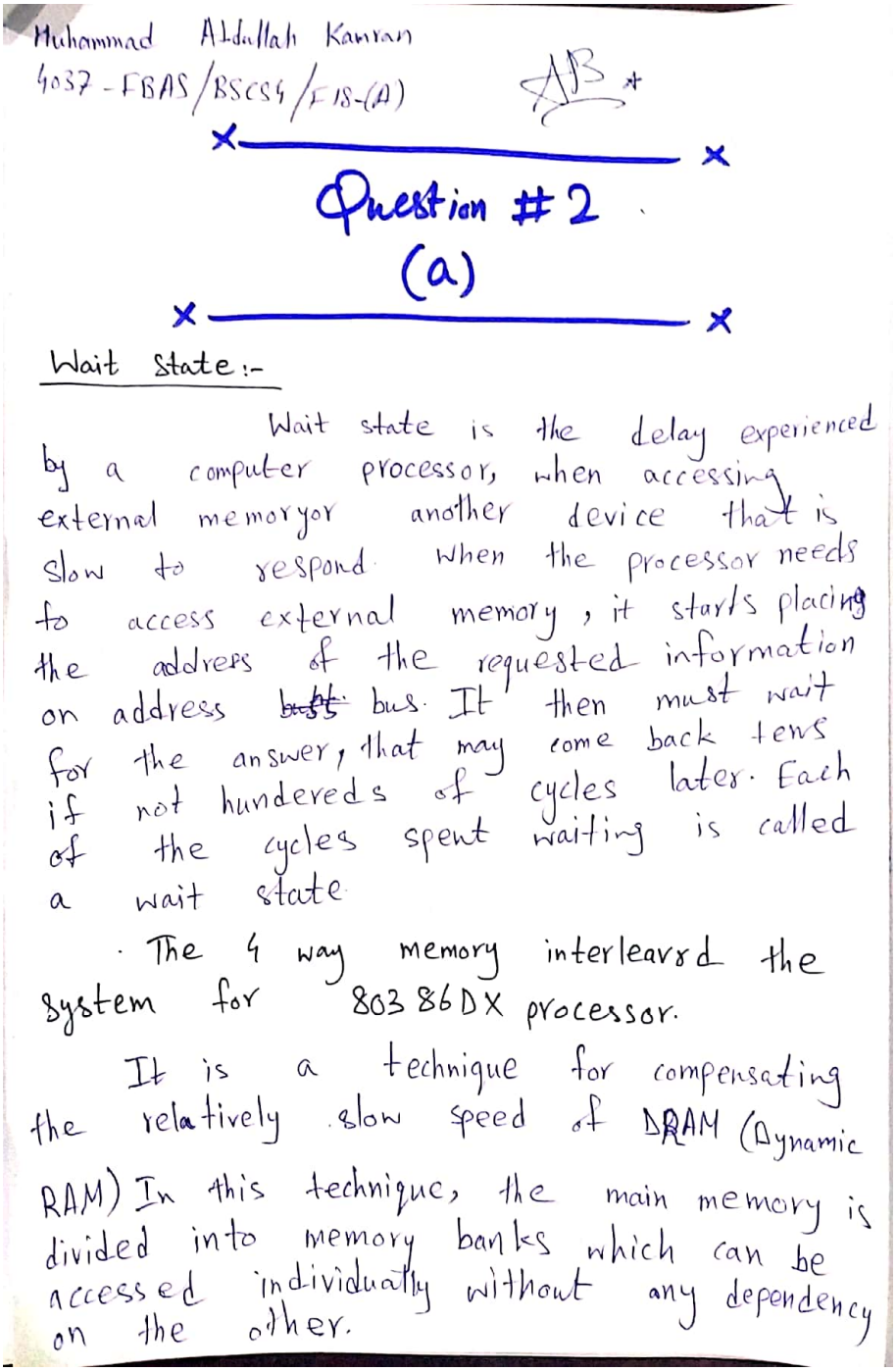
Segment Registor contains a 13bit selector field, a lable selector bit and RPL field.

Table Indicator
0: Global Descriptor Table
1: Local Descriptor Table

Q2.

Insert Pictures of Answer Sheet Here

(A)



Muhammad Abdullah Kamran
4037-FBAS/BSCS4/FIS-(A)

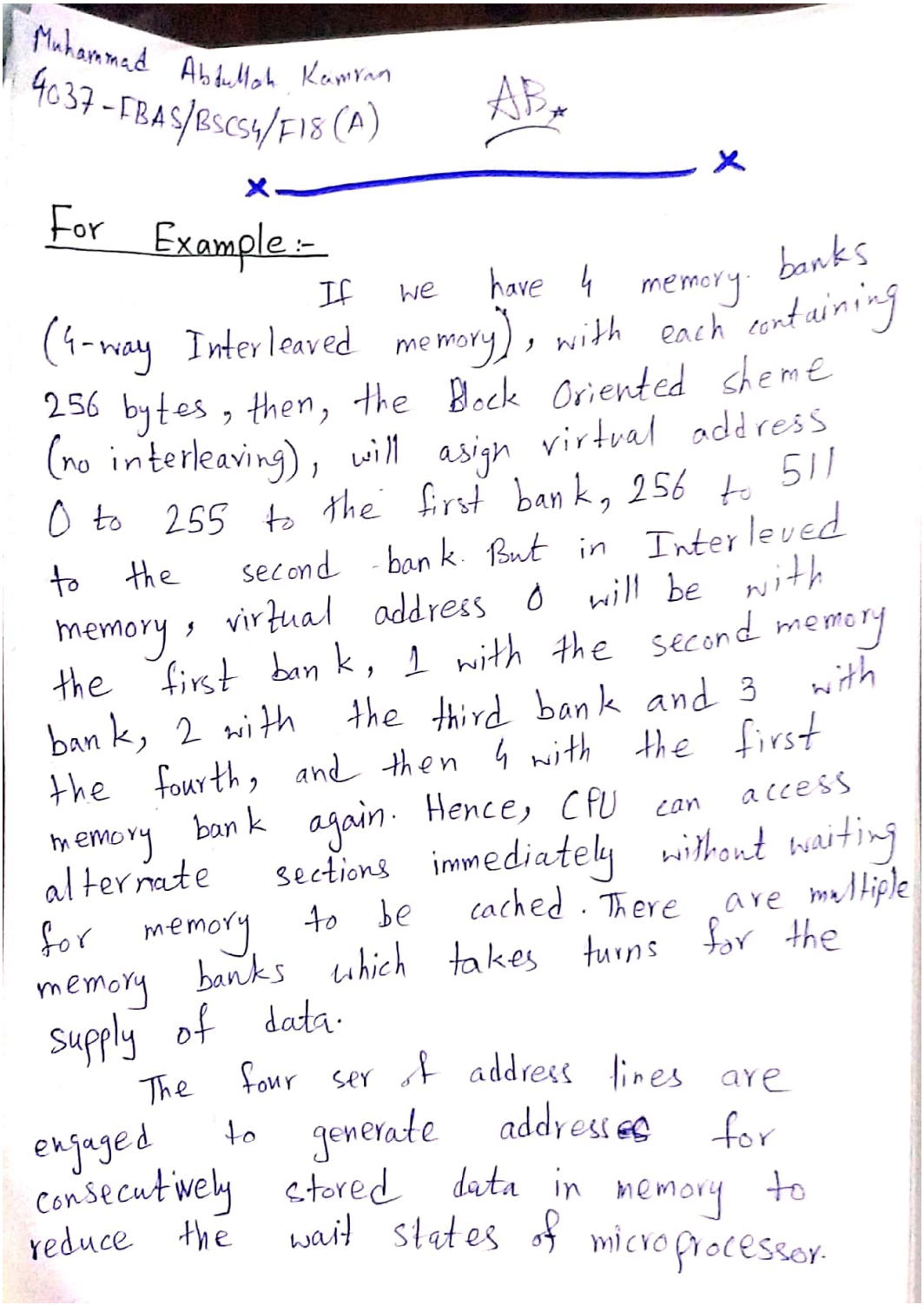
Question # 2
(a)

Wait State :-

Wait state is the delay experienced by a computer processor, when accessing external memory or another device that is slow to respond. When the processor needs to access external memory, it starts placing the address of the requested information on address ~~bus~~ bus. It then must wait for the answer, that may come back tens if not hundreds of cycles later. Each of the cycles spent waiting is called a wait state.

· The 4 way memory interleavrd the system for 80386DX processor.

It is a technique for compensating the relatively slow speed of DRAM (Dynamic RAM) In this technique, the main memory is divided into memory banks which can be accessed individually without any dependency on the other.



Muhammad Abdullah Kamran
4037-FBAS/BSCS4/F18(A)

AB

For Example:-

If we have 4 memory banks (4-way Interleaved memory), with each containing 256 bytes, then, the Block Oriented sheme (no interleaving), will asign virtual address 0 to 255 to the first bank, 256 to 511 to the second bank. But in Interleved memory, virtual address 0 will be with the first bank, 1 with the second memory bank, 2 with the third bank and 3 with the fourth, and then 4 with the first memory bank again. Hence, CPU can access alternate sections immediately without waiting for memory to be cached. There are multiple memory banks which takes turns for the supply of data.

The four ser of address lines are engaged to generate addresses for consecutively stored data in memory to reduce the wait states of microprocessor.

Muhammad Abdullah Kamran
4037-FBAS/BSCS4/F18 (A)

AB*

Wait state is a delay experienced by a microprocessor when accessing external memory or another devices that is slow to respond, the vice versa also come into scenario. Now, to be able to access slow memory the microprocessor must be able to delay the transfer untill the memory access is complete. One way is to increase the microprocesor clock period by reducing the clock frequency. Some microprocessor provide a special control input called ready, to allow the memory to set its own memory cycle time. If after sending an address out, the microprocesors does not recieve a READY input from memory, it enters a wait state for as long as the READY line is in 0 state. When the memory access is completed the READY goes high to indicate that the memory is ready for spcified transfer.

(B)

Muhammad Abdullah Kamran
4037 - FBAS/BSCS4/F18(A)

Question #2

(b)

(a)

Comparing unsigned AX to BX :-

```
mov  Large, bx
cmp  ax, bx
jna  Next
mov  Large, ax
Next:
```

(b)

1st Method :-

```
.data
binNum 1   Byte   11111111b
binNum 2   Byte   00000001b

.code
main   Proc
move   a1, binNum 1     ; AL = 0FFh
sub    al, binNum 2     ; AL = FFh, CF = 01
sub    al, binNum 2     ; AL = FEh, CF = 0
```

Muhammad Abdullah Kamran
4037-FBAS/BSCS 4/F18(A)

AB*

2nd Method :-

Compare Instruction:

Destination > Source	CF = 0
Destination = Source	CF = 0

• code

```
move  ah  42h
move  al  32h
cmp ah, al
```

• code

```
main  proc
mov  cx, '1'
mov  count, '2'

push  cx
push  count

pop   cx
pop   count

mov  dx, cx
mov  ah, 2
int  21h
```

Muhammad Abdullah Kamran ③
4037-FBAS/BSCS4/F18 (A) AB

```
mov dx , count
mov ah, 2
int 21h

mov ah , 4ch
int 21h

main endp
int 21h
```

(d)

Initializing String with your full Name:-

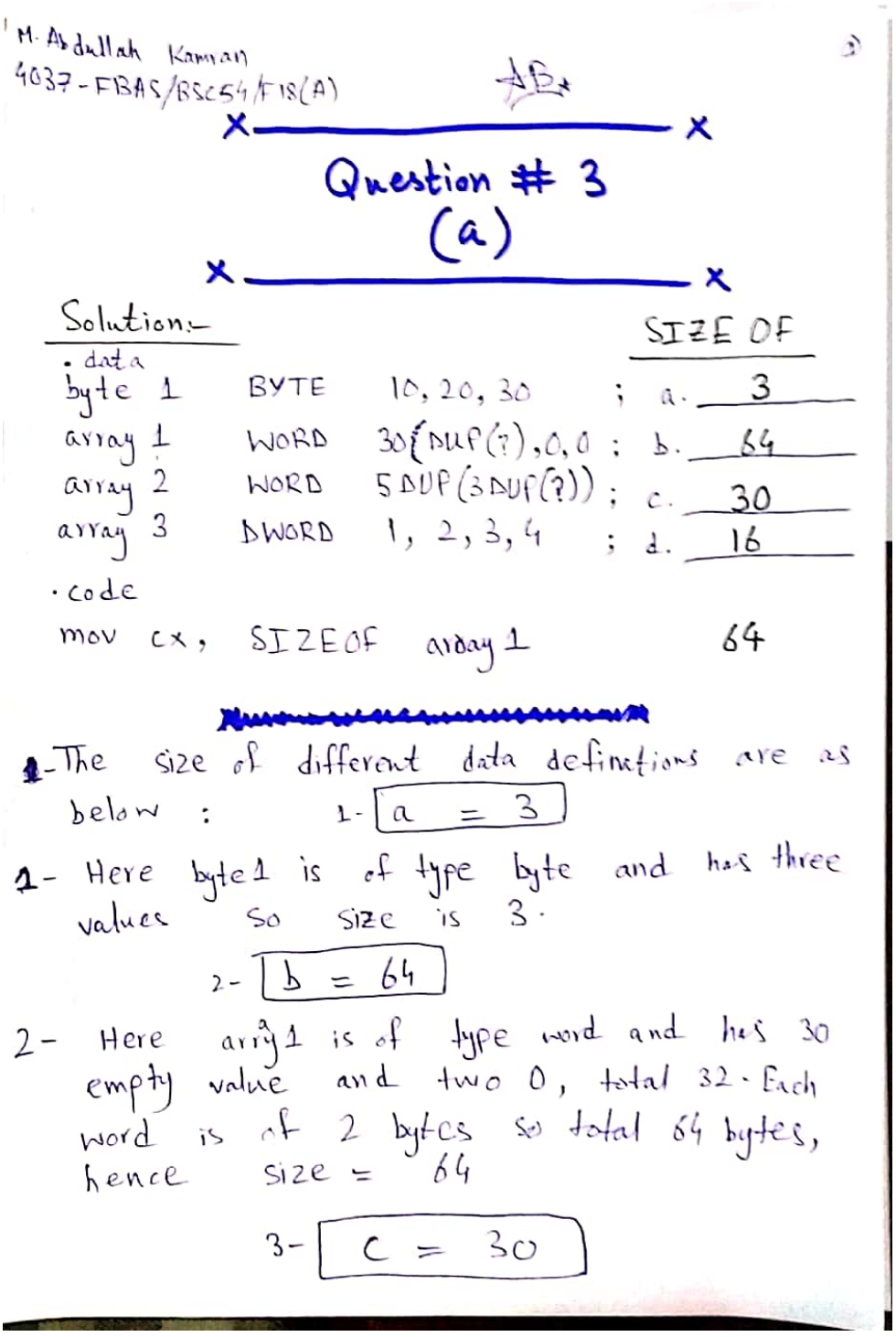
```
name db 'Muhammad Abdullah Kamran', 0
```

Here, we have intialized a string name including my full name.

Q3.

Insert Pictures of Answer Sheet Here

(A)



M. Abdullah Kamran
4037-FBAS/BSCS4/F18(A)

Question # 3
(a)

Solution:-

```
.data                                          SIZE OF
byte 1    BYTE    10, 20, 30          ; a. 3
array 1   WORD    30 DUP(?),0,0       ; b. 64
array 2   WORD    5 DUP(3 DUP(?))     ; c. 30
array 3   DWORD   1, 2, 3, 4          ; d. 16
.code
mov cx, SIZEOF array 1                 64
```

1- The size of different data definations are as below : 1- a = 3

1- Here byte 1 is of type byte and has three values so size is 3.

2- b = 64

2- Here array 1 is of type word and has 30 empty value and two 0, total 32. Each word is of 2 bytes so total 64 bytes, hence size = 64

3- c = 30

Muhammad Abdullah Kamran :-
4037-FBAS/BSCS4/F18(A) :-

3- Here array2 is of type word and has 5x3 empty values, total 15. Each word is of 2 bytes so total 30 bytes, hence size = 30.

4- d = 16

4- Here array3 is of type dword and there are 4 elements. Each dword is of 4 bytes so total 16 bytes, Hence size = 16.

(B)

```
DOSSEG
.MODEL SMALL
.STACK 100H
.DATA
   array DB 20 DUP('$')
 var DB ?
 str1 db 'The number is in list.$'
 str2 db  'Number not found.$'

.CODE

MAIN PROC

   MOV AX, @DATA
   MOV DS, AX

       MOV si,offset array
       mov bx,0
  Label1:
  inc bx
  Mov ah,1
  Int 21h
  cmp al,13
  JE Label2

       MOV [si],al
       inc si
              jmp Label1

   Label2:
```

```
        dec bx


        ;mov dx,offset array
        ;mov ah,9
        ;int 21h


 mov ah,1
int 21h
mov var,al




mov si, offset array
mov cx, bx

search:

mov dx, [si]
cmp dx, var
JE result
inc si

loop search

  lea dx, str2
  mov ah,9
  int 21h
```

```
    MOV AH, 4CH
    INT 21H


result:
cmp cx,10
jl print1
jg  print2


print1:
 mov dx,cx
 add dx,48
 mov ah,2
 int 21h

MOV AH, 4CH
    INT 21H

print2:
 lea dx, str1
  mov ah,9
  int 21h


    MOV AH, 4CH
    INT 21H

    MAIN ENDP
    END MAIN
```