19/72

BIRZEIT UNIVERSITY

College of Engineering and Technology

Computer Science Department

Name [redacted] Number [redacted]

Comp431 Midterm Exam Fall 16/17

[1] In Dynamic(variable) regions memory management (MVT), the memory status looks like:

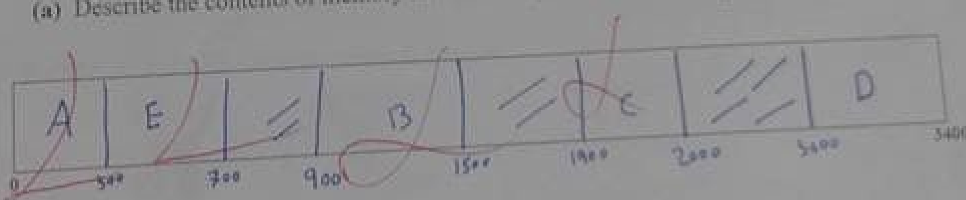
Given the following memory requests in the given order:

- Process E starts and requests 400 memory units.
- Process A requests 500 more memory units.
- Process C exits.
- Process F starts and requests 300 memory units.
- Process B exits.
- Process G starts and requests 700 memory units.

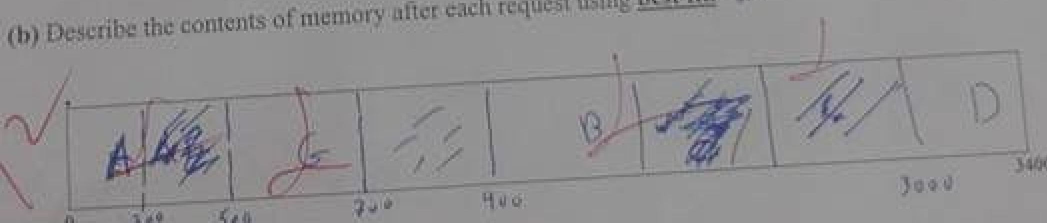
<u>*NOTE:*</u>

- For a growing process, if the current state cannot accommodate a request, reallocate the process.
- If the request can't be accommodated, just state that.

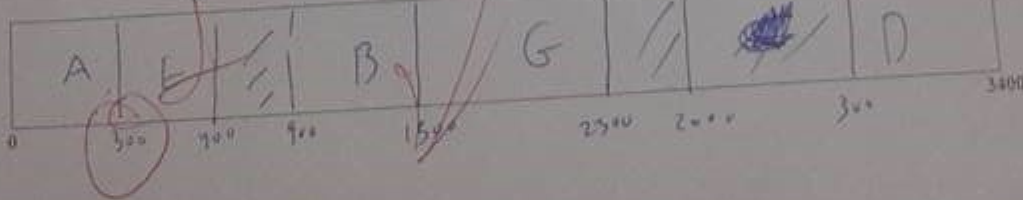
(a) Describe the contents of memory after each request using **<u>first fit</u>**.



(b) Describe the contents of memory after each request using **<u>best fit.</u>**



(c) Describe the contents of memory after each request using **<u>worst fit.</u>**



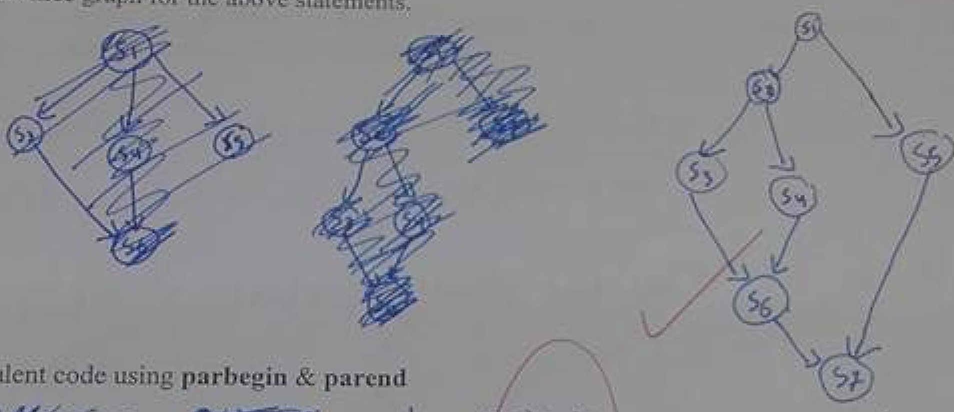
[2] Given the following statements which will be executed in the same order:

S1 : int x, y, z, w ;
S2 : x = 1 , y=10;
S3 : x += 10;
S4 : y++;
S5 : z=5 ;
S6 : y = x+y;
S7 : w = z - y;

(a) **compute** :

R(S6) = {x, y}
W(S6) = {y}
R(S4) = {y}
W(S4) = {y}

(b) Draw the precedence graph for the above statements.



Write an equivalent code using **parbegin & parend**

```
Parbegin
 S1
 Parbegin
 Parbegin
  S2
 Parend
  S3
  S4
 Parbegin
  S6
 Parend
 S7
```

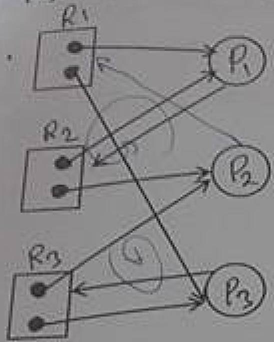
Begin , Parend

Write an equivalent code using **fork & join** structures.

```
Count = 3
S1
fork L1
S2
fork L2
fork L3
S4
S5
L3 : S6
goto L3
L2 : S5
goto L3
L3 : join count
```

confused

[3] Consider the following Resource Allocation graph:



(a) Does the above allocation grapgh contains a deadlock? Explain. If the answer is no, give a possible execution sequence of processes, that is, a safe sequence.

No deadlock because no cyclie with Resurce allocation and their process

→ No dead lock leads to safe sequence for each process.

3 2

safe sequence

) Assume P_2 demands an instance of resource R_1, does the allocation grapgh leads to a deadlock? Explain

yes, will leads to dead lock (cyclie) with their process and Resource allocation.

5

cycle

[4] In a paging system, given that the logical address space (logical program) is 32 pages of 8192 bytes each page, which is mapped to a physical memory of 256 frames.

2^{13} $\rightarrow 2^8$

(a) how many bits we need to address the **logical address**.

~~LA = P%PageSize~~ ~~P = LA/PageSize~~

LA = 13 + 8 = 21 bits

(b) how many bits we need to address the **physical address**.

PA = 21 − 13 = 8 bits

(c) What is the **maximum size** of the program that can run on this machine.

MAX size = 2^{21}

(d) What is the advantage of allowing more than one entry in the page table (each entry from diffe… process) to point to the same frame in memory.

Sharing Paging

[5]

(a) Explain the difference between preemptive and nonepreemptive SJF scheduling algorithm.

Preemptive

1) switching from running To ready
2) switching from waiting To ready

Nonepreemptive

1) switch from running to waiting
2) Exit (termente)

(b) Define starvation.

it the problem caused when each process want to finished their execution (their time (cpu Burst)), it problem in SJF scheduling ~~multivel feedback~~.

(c) can starvation ocuur in nonepreeptive SJF scheduling? Explain.

Yes, stravation is problem in executed nonepreeptive in short Job first because each process stravated with another process (cpu Burst) to finished their proces

) One way to handle starvation is "Aging", What is Aging?

move up the process in multivel feedba[ck] that help to suffers starvaion.

BIRZEIT UNIVERSITY

College of Information Technology

Computer Science Department

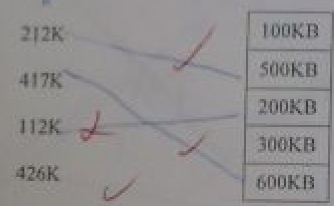
Name [redacted] Number [redacted]

Comp431 Midterm Exam Fall 13/14

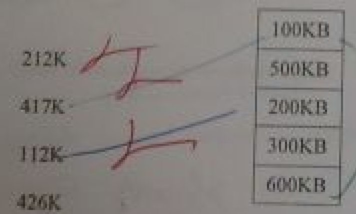
[1] In **dynamic partitions** memory management, given the *holes* :
100KB, 500KB , 200KB , 300KB , 600KB (in order)

Given the processes 212K , 417K , 112K , 426K (in order) . **Draw** arrows showing where you place each process in the following allocation algorithms.

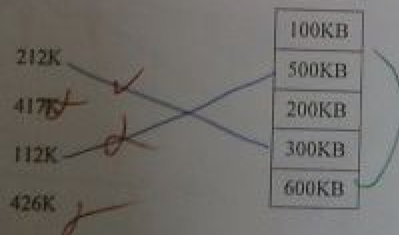
First-Fit



Worst-Fit



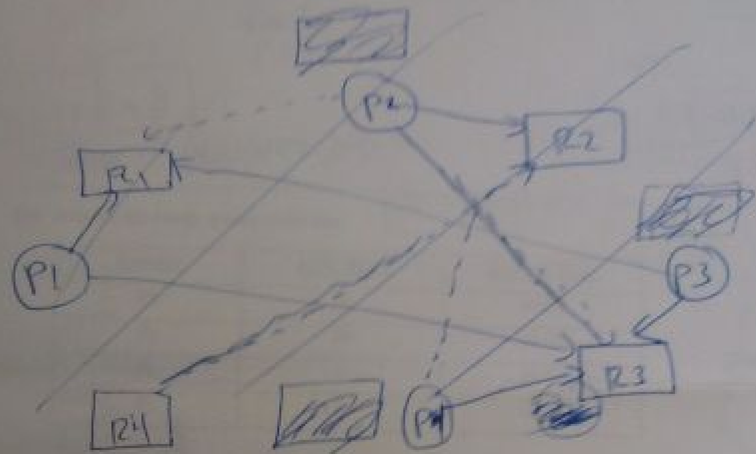
best-Fit



[2] A system is composed of four processes, p_1, p_2, p_3 and p_4 and three types of resources, R_1, R_2, R_3. There is one unit of each resource type available.

- p_2 assigned a unit of R_1 and a unit of R_4
- p_4 assigned a unit of R_1 and requests one unit of R_3
- p_3 requests one unit of R_1
- p_1 requests a unit of R_1 and a unit of R_3
- p_3 requests a unit of R_1 and a unit of R_3

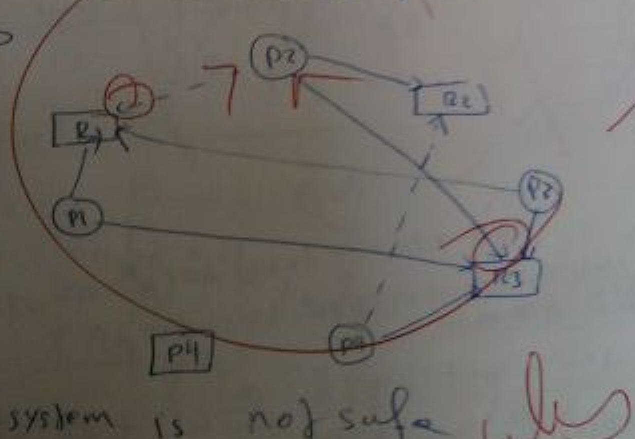
(a) Draw the Resource Allocation Graph that represents the system state.



(b) Is the system is safe ? If not, which of the processes are deadlocked?

safe system: request available state resource and the system decide if it is the resource release place.

حل السؤال
(a)



15

(b) the system is not safe

ليه

[3] (a) If the LA is 20 bits long and given the LA = (00000101000100000111)$_2$, and page size = 4096 word, and given the page table :

- What is p and d without using the / and % operations.
- compute PA

10
150
840
122
40
20
10
4096
.
100

P = 8

8 (page)	12 (offset)

00000101	000100000111

p = 8 bit

PA = s*F+d

PA = 82055

(b) Assume the ready queue looks like:

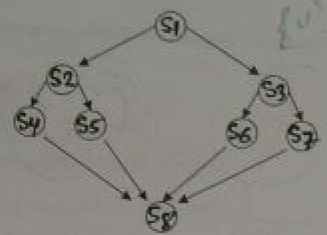
Process	CPU Burst	Arrival Time
P_1	30	10:00
P_2	20	10:08
P_3	3	10:17
P_4	8	10:22

Compute the AWT in the preeptive SJF scheduling algorithm.

Arrive	cpu burst
10:00	30
10:08	20
10:17	3
10:22	8

AWT = (71 − 10 − 30) + (41.08 − 10.08 − 20) + (21.17 − 10.22 − 8) + (13.17 − 10.17 − 3) / 4 = (31 + 11 + 0 + 2.95) / 4 = 44.95 / 4 = 11.2

[4] Given the precedence graph:



(a) Write an equivalent code using **parbegin** & **parend**

(b) Given the following statements which will be executed in the same order:

S_1 : int x=1 , y=100;
S_2 : x += 10;
S_3 : y++;
S_4 : int z;
S_5 : z = x+y;
S_6 : cout << z+x;

(a) compute :

The write set , $W(S_6)$ =

The read set , $R(S_5)$ =

[5] (a) Given the only data structure declaration:
int semaphore S = 1;

do

Define the *wait(S)* and *signal(S)* instructions.

wait (full)
wait(mutex)
remove an item
signal (mutex)
signal (empty)

s.value = s.value + 1

s.value = s.value - 1

(b) Show that if the *wait(S)* and *signal(S)* semaphore operations are not executed **atomically** ,then mutual exclusion may be violated.

wait (full
wait(mutex
remove

800-64=

(c) A program of 800 bytes size. For each of the following memory management , Compute the amount **internal fragmentation** if there is any :

Paging with page size 2^6 (64) 64 byte

$\frac{800}{2^6}$ = 12.5 page

64-32 = 32

MFT with 3 partitions 1200 , 900 , 5000 and **best available fit** only algorithm.

900 - 800 = 100 byte

500

Segmentation with equal segments size = 256 bytes

800 % 256 =

256 - 32 = 224 byte

[6] (a) On IBM 370 OS with **paging** memory management, the logical address is 24 bits, 11 bits for the page number and 13 bits for the offset. The page table entry is 3 bytes long containing 20 bits for the frame number, and 4 bits for the **legal/illegal** bit, **V/I** bit, **dirty** bit, and **reference** bit.

1. What is the page size.

page # $= 2^{11}$ 2^{13} $2^{11}-1$ entries

2. What is the size of memory.

$20+13=33$

$(2^{11})(2^{13}) = 2^{24}$ 2^{33}

page | offset

3. What is the size of page table.

$2^{11} \cdot 4 = (2^{10})(2)(4)$

$= (2^{10})(2)(4)$

$= 8K$ $= (8)(2^{10})$ bytes

$= 8K$

(b) A demand paging system with **4-levels** page table kept in **memory**.
Assume memory access = 10 mics

(i) Compute the time to access(reference) an instruction.

(ii) If an associative registers table is added to the system with **hit ratio** 98%. Compute the EAT given that the lookup time in the associative registers is 1 mics.

$EAT = h(t+m) + (1-h)(t+5m)$

$h(t+m) + (1-h)(t+?m)$

$Time = 0.98(1+10) + 0.02(1+50)$

$= 0.98(11) + 0.02(51)$

$= 10.78 + 1.02$

$= 11.8$ mics

BIRZEIT UNIVERSITY
College of Information Technology
Computer Science Department

Name: Ayah Belal
Comp331

Number: 11XX985

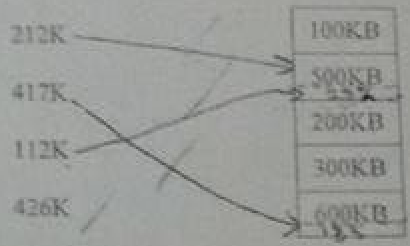
Midterm Exam

Fall

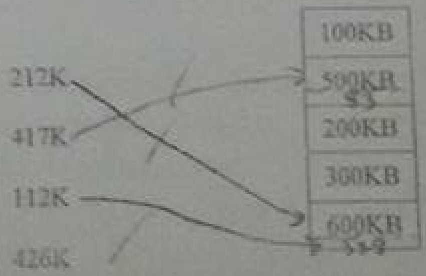
[1] In dynamic partitions memory management, given the *holes* :
100KB, 500KB , 200KB , 300KB , 600KB (in order)

Given the processes 212K , 417K , 112K , 426K (in order) . **Draw** arrows showing where you place each process in the following allocation algorithms.

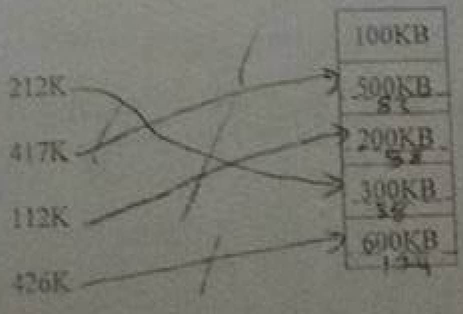
First-Fit



Worst-Fit



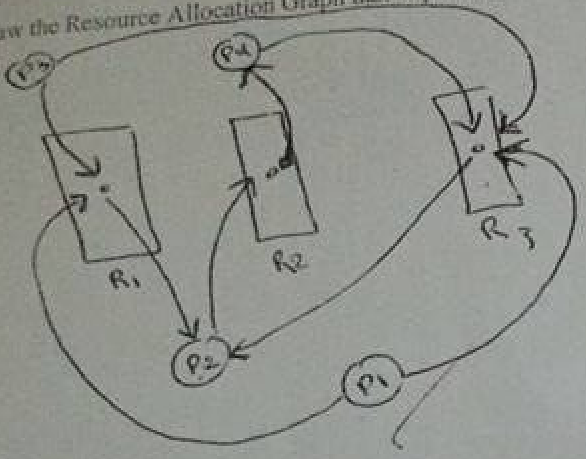
best-Fit



[2] A system is composed of four processes, p_1, p_2, p_3, and p_4, and three types of resources, There is one unit of each resource type available.

- p_2 assigned a unit of R_1 and a unit of R_3. ×
- p_4 assigned a unit of R_2 and requests one unit of R_3. ×
- p_2 requests one unit of R_2. ×
- p_1 requests a unit of R_1 and a unit of R_3. ×
- p_3 requests a unit of R_1 and a unit of R_3.

(a) Draw the Resource Allocation Graph that represents the system state.



b) Is the system is safe ? If not, which of the processes are deadlocked?

safe state ⇒ safe sequencess يرتبوا تكون
: sequencess of process.

it is not a safe state, there is a deadlock
the deadlock is :- P2 → R2 → P4 → R3 → P

if the process 2 does not hear then ~~the~~ ~~deadlock~~ will not be a deadlock.

If the LA is 20 bits long and given the LA = (00000101000100000111)$_2$, and page size = 4096
and given the page table :
What is p and d without using the / and % operations.
compute PA

0	10
1	150
2	840
3	122
4	40
5	20
	10
	4096
	100

page table

1) page size = 4096 = 2^{12}

$\Rightarrow$ d = 12

LA = P + d $\Rightarrow$ 20 = P + 12

$\Rightarrow$ P = 8

LA: | 8 bit (P) | 12 bit (d) |

| 5 (P) | 263 (d) |

2) PA = S * F + d

PA = 4096 * 20 + 263 = 2^{12} * 20 + 263

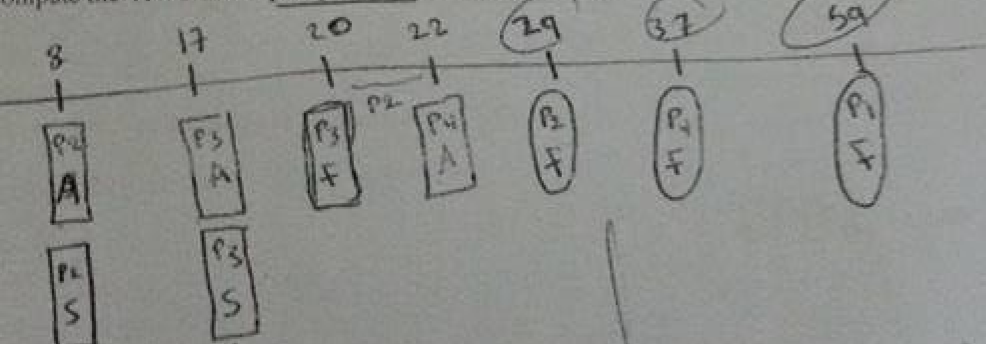
= 82183.

b) Assume the ready queue looks like:

Process	CPU Burst	Arrival Time
P_1	30 22	10:00 0
P_2	20 9 7 0	10:08 8
P_3	7 0	10:17 17
P_4	8 0.	10:22 22

61

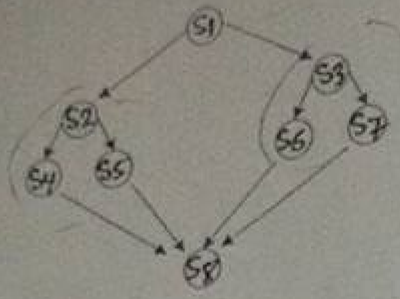
Compute the AWT in the preeptive SJF scheduling algorithm.



AWT = $\frac{(59-0-30) + (29-8-20) + (20-17-3) + (\ldots}{4}$

AWT = $\frac{29+1+0+7}{4} = \frac{37}{4} = 9.25$

[4] Given the precedence graph:



(a) Write an equivalent code using **parbegin & parend**

```
S1;
parbegin
  begin
    S2;
    parbegin
      S4;
      S5;
    parend
  end
  begin
    S3;
    parbegin
      S6;
      S7;
    parend.
  end
parend
S8;
```

(b) Given the following statements which will be executed in the same order:

S_1 : int x=1 , y=100;
S_2 : x += 10;
S_3 : y++;
S_4 : int z;
S_5 : z = x+y;
S_6 : cout<<z+x;
printf.

(a) compute :

The write set , $W(S_6)$ = ∅ or { }.

The read set , $R(S_6)$ = { x , y }

...es the only data structure declaration:
...nt semaphore S = 1;

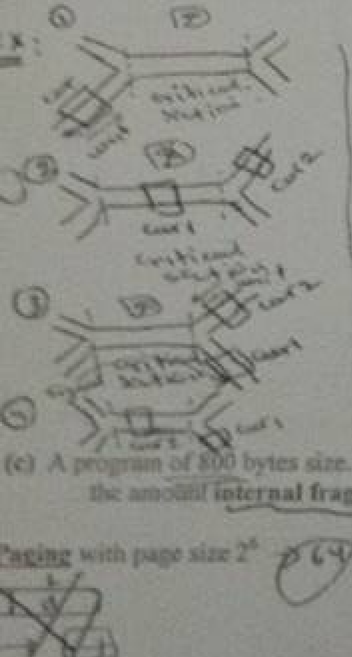
Define the *wait(S)* and *signal(S)* instructions.

wait → entry section
signal → exit section

wait(S)
S = S - N
}

signal (S)
{
S = S+1
}

(b) Show that if the *wait(S)* and *signal(S)* semaphore operations are not executed **atomically**, then mutual exclusion may be violated.



atomically

atomically

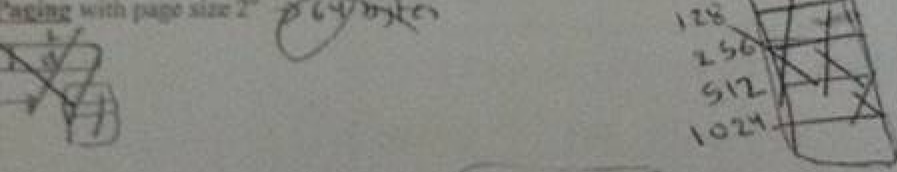
critical section

signal, wait

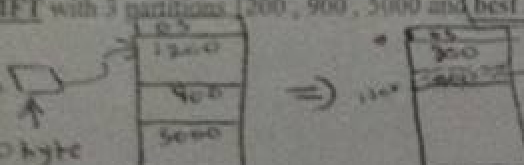
critical section

(c) A program of 800 bytes size. For each of the following memory management, Compute the amount **internal fragmentation** if there is any :

Paging with page size 2^6 = 64 bytes



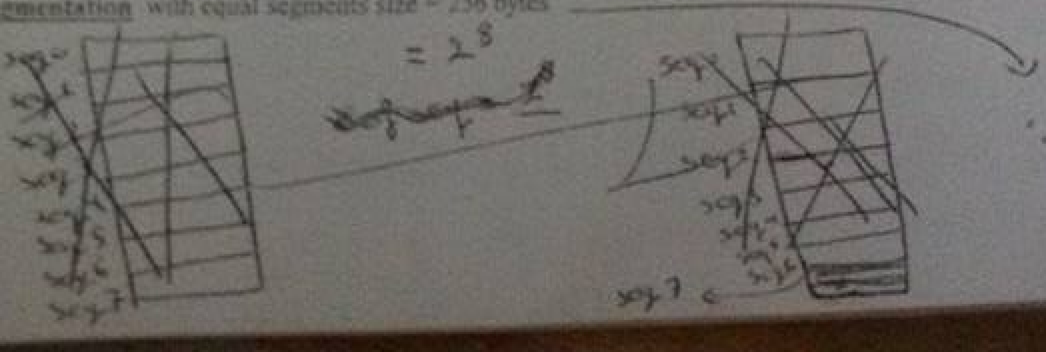
MFT with 3 partitions 1200, 900, 5000 and **best available fit** only algorithm.



internal fragmentation = 400

Segmentation with equal segments size = 256 bytes

$= 2^8$



[6] (a) On IBM 370 OS with paging memory management, the logical address is 24 bits, 11 b... the page number and 13 bits for the offset. The page table entry is 3 bytes long containing ... bits for the frame number, and 4 bits for the legal/illegal bit, V/I bit, dirty bit, and ref...

1. What is the page size. 2^{d}

$2^{11}-1$

2. What is the size of memory. (P memory)

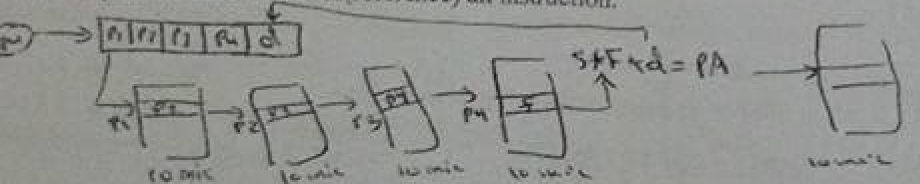
P size of memory $= 2^{13} * 20$
$= 163840$

3. What is the size of page table.

size of page table $= 2^{P} * 7$ bytes
$= 2^{11} * 7$
$= 14336$

(b) in paging system with 4-levels page table kept in memory.
Assume memory access = 10 mics

(i) Compute the time to access(reference) an instruction.



step = 4 ⇒ memory needed = step + 1 = 5

(ii) If an associative registers table is added to the system with hit ratio 98%. Compute the EAT given that the lookup time in the associative registers is 1 mics.

$t = 1$ mics, $m = 10$ mics, $h = ,98$

$EAT = h(t+m) + (1-h)(t+5m)$
$= ,98(1+10) + (1-,98)(1+5*10)$
$= 10,78 + 0,02 * 51$
$= 10,78 + 1,02$
$= 11,8$

86

Second ~~Second~~

Second

بيرزيت

Birzeit University - Computer Science Dept.

Name الطراونة Number 960482 ...

Comp 431 Exam. #2 29/01/2001

Question (1) : Define the following :

(a) Dirty Bit : a bit added to the page table entry which indicates wether the page is modified or not

(b) Valid/Invalid Bit : a bit add to page table entry which ... legal address and if referred page is in memory

(c) PTBR : page table base register contains the address of begining of page table in memory

(d) Page Fault Rate

the percentage of time of that the page will not be in the ~~associative register~~ memory

page table

Question (2)
In Paging memory management, if virtual address is 20 bits long with page ... 256 bytes. If physical memory is 64MB and given the following page table ...

(a) What is the maximum size of programs executed?

| 12 | 8 |

$2^{12} \times 2^{8} = 2^{20} = 1\,MB$

virtual

(b) Compute the size of the page table?

$2^{26} \rightarrow 18$ bits $\rightarrow$ need 2 bytes

$2^{12} \times 2 = 2^{13} = 2K$

(1537)

(c) Given the virtual address in Octal $(3025)_8$. Compute the Physical Address (...

011 000 010 101

$d = 00010101 = 21$

$P = 0110 = 6$

1537

$Ph.A = 66 \times 256 + 21$

IBM 370 OS with paging memory management, the logical address is 24 bits, 11 bits for the page number and 13 bits for the offset. The page table entry is 3 bytes long containing 20 bits for the frame number, and 4 bits for the **legal/illegal bit, V/I bit, dirty bit, and reference bit.**

1. What is the page size.

page size = 2^d

$\Rightarrow d = 13 \Rightarrow$ page size $= 2^{13} = 8192$

2. What is the size of memory.

Memory size = 2^{LA}

$\Rightarrow 2^{24} = 16777216$ bit

3. What is the size of page table.

Size of page table = $2^{11} * 20$

$= 40966$ bit

$= 2^p *$ entry bits $= 40960$

In A ~~A demand~~ **paging** system with **4-levels** page table kept in **memory**,

ssume memory access = 10 mics

Compute the time to access(reference) an instruction.

The Time = $10 * 4 = 40$ unit

↳ 40 mics

$10 \times 5 = 50$

If an associative registers table is added to the system with hit ratio 98%. Compute the EAT given that the lookup time in the associative registers is 1 mics.

EAT = $h * (t + m) + (1 - h) * (t + 5m)$

$= 0.98(11) + 0.02 * 51$

$10.78 + 1.02$

EAT = 11.8 unit

↳ 11.8 mics