Operating Systems Concepts

Chapter 7

Memory management

Prepared by Assist . Prof .

Imad Matti

AL-mamoon University College / Computer Science Department

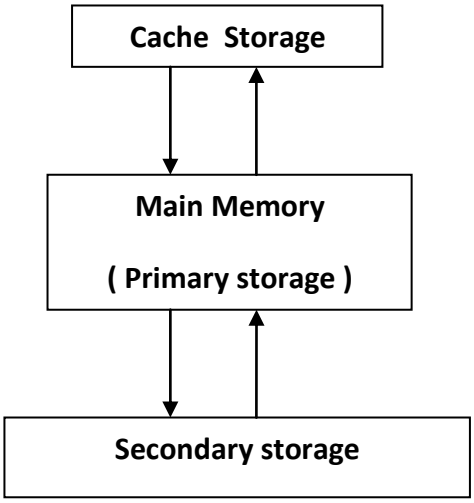
2018

Chapter 7

Memory Management

7. Introduction

Programs and data need to be in main storage in order to be executed, And if they do not needed immediately, they may be kept on secondary storage media such as tapes or disk until needed and then brought into the main storage for execution.



(Storage Organization)

Logical (virtual) **address**: address generated By CPU.

Physical address: address seen by the memory unit.

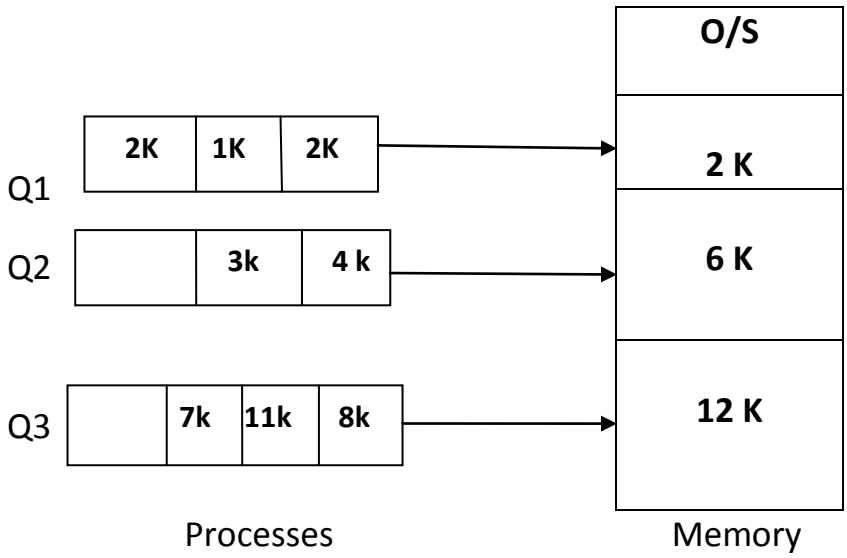
7.1 Multiple Partition Allocation

There are two memory management schemes:

7.1.1 Multiple contiguous fixed partition (MFT)

The memory is divided into a number of fixed sized partitions. Each one contain exactly one process.

Example 1: Consider the following:



Disadvantages:

The problem with MFT is **internal** and **external fragmentation**.

The solution to this problem is **MVT**.

7.1.2 Storage placement strategies

a. **First**-**Fit** strategy: Allocate job in the **first** hole in memory that is big enough.

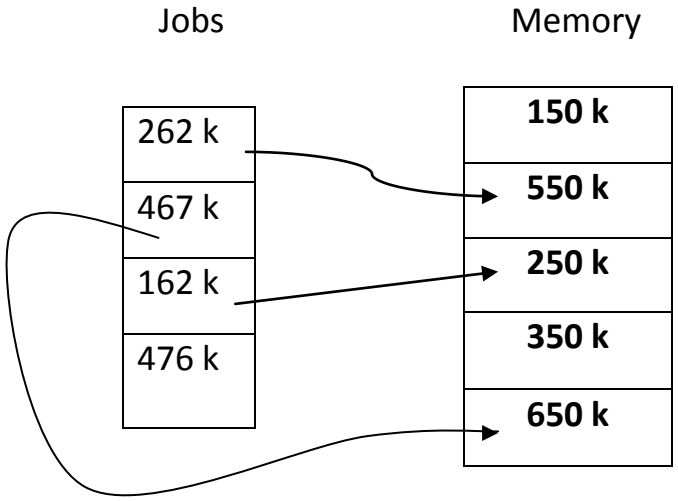
b. **Best**-**Fit** strategy: Allocate job in the **smallest** hole in memory big enough.

c. **Worst**-**Fit strategy**: Allocate job in the largest hole in the memory.

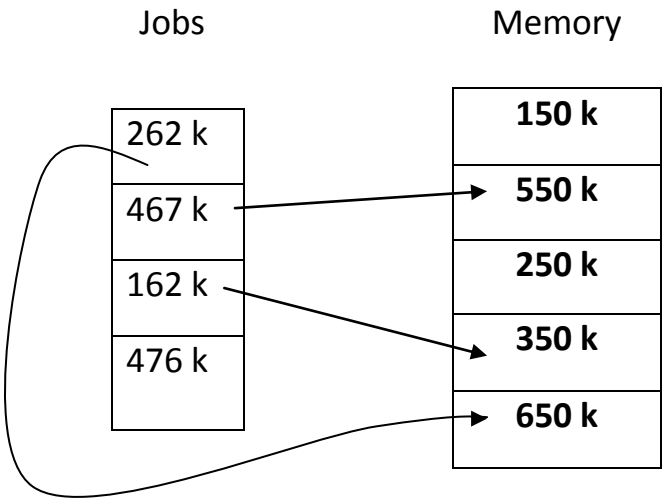
Example 2: Given memory partitions of 150 k, 550 k, 250 k, and 650 k (in order). How would each of the First fit, Worst fit, and Best

fit algorithms place processes (jobs) of 262 k, 467 k, 162 k, and 476 k (in order)?

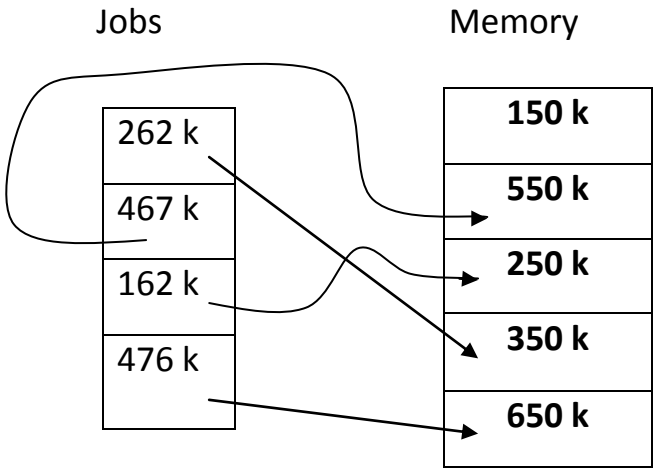
1. First fit:



2.Worst fit:



3. Best fit:



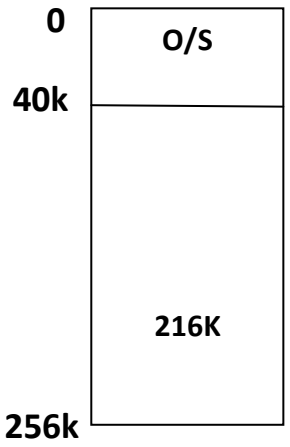
7.1.3 Multiple Variable Partition (MVT)

In MVT, initially all memory is available for user programs and is considered as one large block. When the job arrives and needs memory we search for a hole in memory large enough for this job.

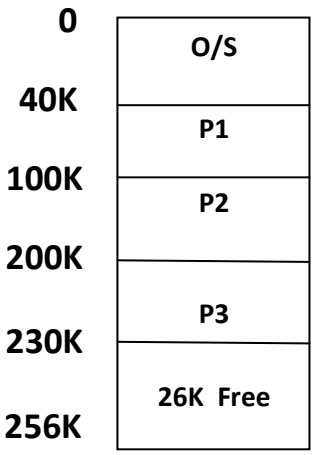
Advantages: minimize internal and external fragmentation.

Example 3: assume we have 256k memory available, and O/S require 40k, with job queue FCFS scheduling, and the following system snapshot, allocate memory to jobs 1, 2, 3, 4, and 5.

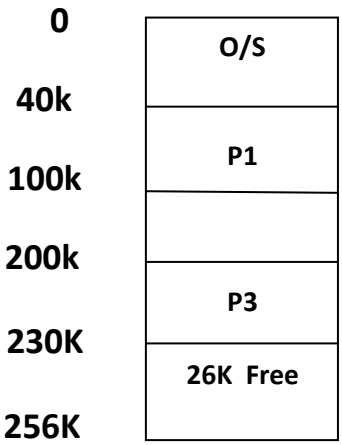
Process	Size	Time
P1	60k	10
P2	100k	5
P3	30k	20
P4	70k	8
P5	50k	15



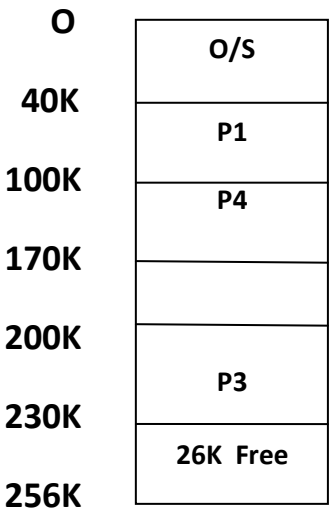
(a)



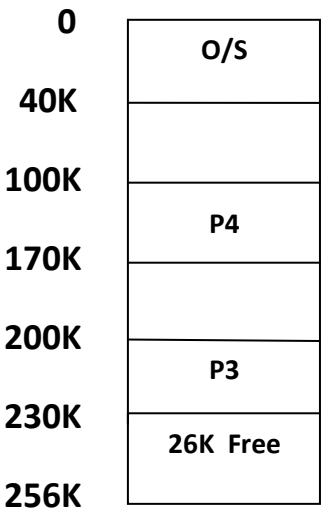
(b)



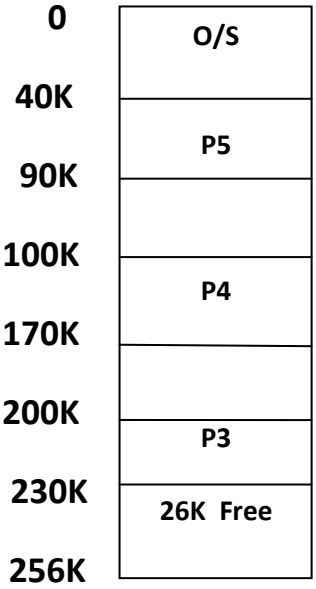
(c)



(d)



(e)



(f)

7.1.4 Fragmentation

There are two types of fragmentation:

1. Internal fragmentation(IF):

some portions of memory are left unused, and it cannot be used by any process.

2. External fragmentation(EF):

Occurs when a region is unused and available but too small for any waiting job.

F = IF + EF

Disadvantage of fragmentation: Memory waste.

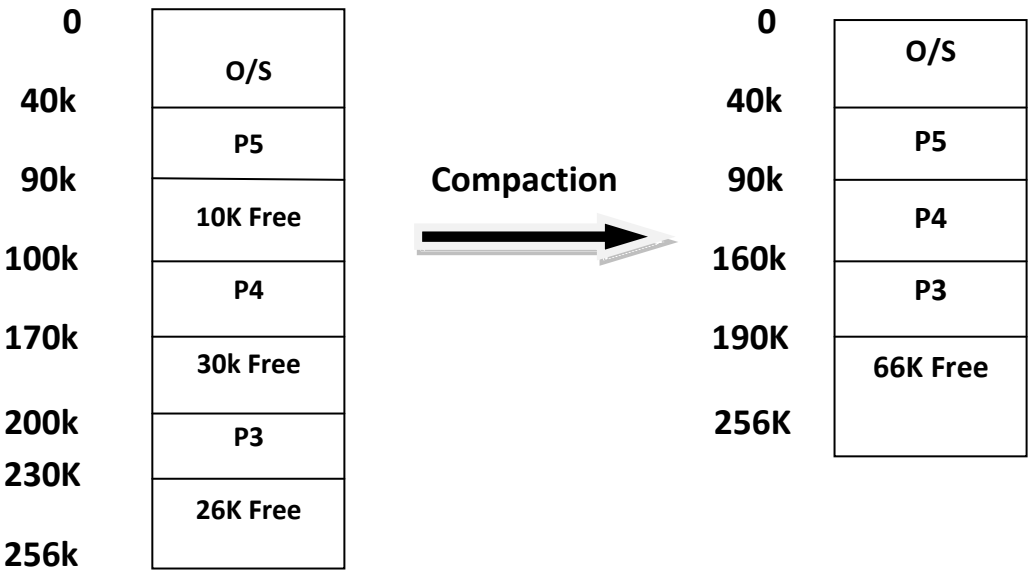
Solutions to the fragmentation problem:

1. **Compaction.**
2. **Paging.**

7.1.5 Compaction

Is to move all jobs towards one end of memory, all holes move in the other direction, producing one large hole of available memory.

Example 4:



7.1.6 Paging Method

In paging the physical memory is broken into blocks called *frames*. Logical memory is also broken into blocks called *pages*. When a process is to be executed its pages are loaded into any available memory frames.

Example 5: A user program consists of 16 instructions, each instruction takes one byte. Suppose page size = frame size = 4 bytes. The physical memory size is 512 bytes.

a. Calculate the number of pages in the logical map.

b. Calculate the number of frames in the physical map.

c. Draw the logical map.

d. Suppose page 1 is allocated in frame 20, and page 2 is allocated in frame 22, calculate the physical addresses of the logical addresses of page 1 and page 2.

e. Draw the physical map.

SOLUTION:

a. Program size = number of program instructions × instruction size.

= 16 × 1 byte = 16 bytes.

Number of pages = program size ÷ page size.

= 16 ÷ 4 = 4 pages in the logical map.

b. Number of frames = physical memory size ÷ page size.

= 512 ÷ 4 = 128 frames in the physical map.

Page number	Logical address	displacement	Instructions
Page 0	0	0	X1
	1	1	X2
	2	2	X3
	3	3	X4
Page 1	4	0	X5
	5	1	X6
	6	2	X7
	7	3	X8
Page 2	8	0	X9
	9	1	X10
	10	2	X11
	11	3	X12

Page 3	**12** **13** **14** **15**	**0** **1** **2** **3**	**X13** **X14** **X15** **X16**

(Logical map)

c.

Page number	Frame number
1	**20**
2	**22**

Page table

Physical address = (frame number × page size) + displacement.

Physical address(x5) = (20 × 4) = 80 + 0 = 80
Physical address(x6) = (20 × 4) = 80 + 1 = 81
Physical address(x7) = (20 × 4) = 80 + 2 = 82
Physical address(x8) = (20 × 4) = 80 + 3 = 83

Physical address(x9) = (22 × 4) = 88 + 0 = 88
Physical address(x5) = (22 × 4) = 88 + 1 = 89
Physical address(x5) = (22 × 4) = 88 + 2 = 90
Physical address(x5) = (22 × 4) = 88 + 3 = 91

Physical address	Displacement	instructions	Frame number
			0
			……..
80 **81** **82** **83**	**0** **1** **2** **3**	**X5** **X6** **X7** **X8**	**20**
			……..
88 **89** **90** **91**	**0** **1** **2** **3**	**X9** **X10** **X11** **X12**	**22**
			……
			127

d. Physical map

7.1.7 Segmentation

Segmentation: is memory management scheme that supports user view of memory.

Example 6: Consider a program consists of five segments S0 = 600kB, S1 = 14kB, S2 = 100kB, S3 = 580kB, S4 = 96Kb.

Assume at that time the available free space partitions of memory are:

1200-1850kb, 50-160kb, 220-250kb, 2500-3200kb

a. Allocate space for each segment in the order given above.

b. Draw the logical, physical maps and segment table.

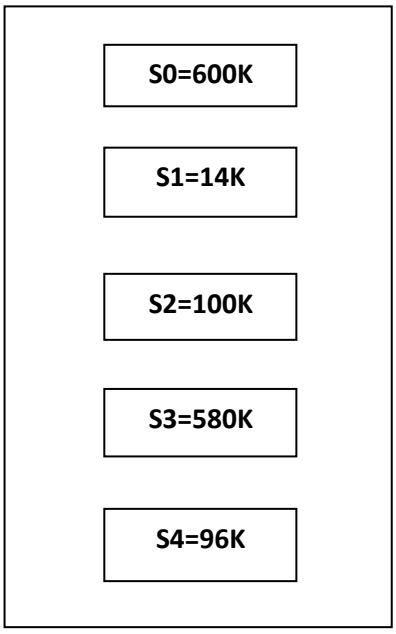
c. Calculate the external and internal fragmentations, where if the remainder of partition < 10 bytes leave it as internal fragmentation and use the BEST-FIT strategy to allocate space for each segment.

d. What are the physical addresses in memory for the following logical addresses:

(1) 0.580 (2) 1.17 (c) 2.96 (d) 4.112 (e) 3.420

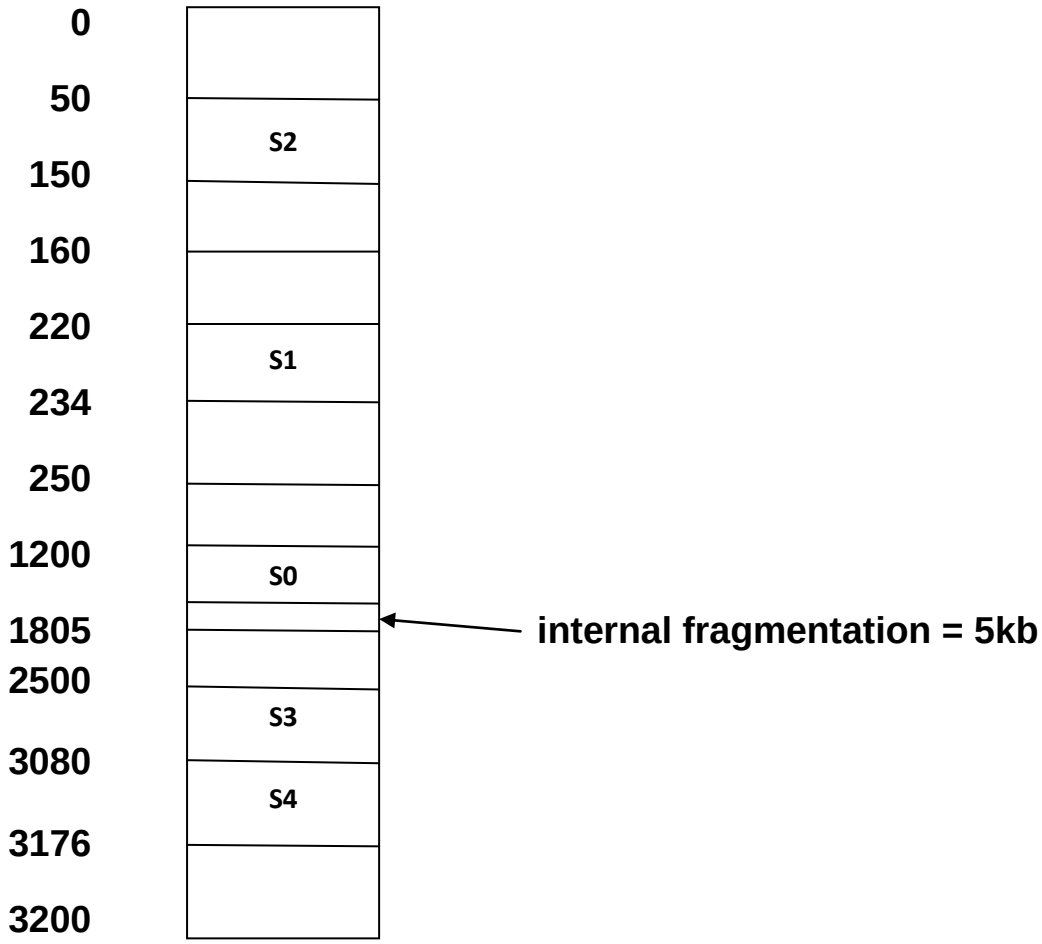
<u>The solution</u>:

The logical map and segment table before allocation are:



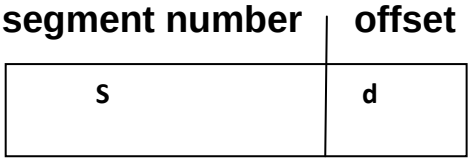
Segment number	Limit register	Base register
0	600	1200
1	14	220
2	100	20
3	580	2500
4	96	3080

The segment table



Physical memory

d.



if d < limit

then:

physical address = base address + d

else error (no physical address)

(1) 0.580

S=0 d=580

d < limit

580 < 600 --> true

Physical address = 1200 + 580 = 1780

(2) 1.17

S=1 d=17

d < limit

17 < 14 --> false --> error (no physical address)

(3) 2.96

S = 2 d = 96

d < limit

96 < 100 --> true

Physical address = 50 + 96 = 146

(4) and (5) home work

Chapter 7 Questions

Q1: Given a snapshot of a memory management table and an input queue as shown in the following: (وزاري ٢٠١٦)

Memory Management Table
10 KB
30 KB
20 KB
40 KB
50 KB

Process Name	Input Queue or (process requirements)
P1	10 KB
P2	20 KB
P3	15 KB
P4	30 KB
P5	25 KB

Show how First Fit, Best Fit, and Worst Fit algorithms would allocate the processes in the input queue in the spaces shown in the memory management table.

Q2: A computer system with a memory consists of the following hole sizes in the following memory order: 10 KB , 4 KB , 20 KB , 18 KB , 7 KB , 9 KB , 12 KB, 15 KB.

Which hole is taken for successive block request of program size request of 14 KB , 9 KB , 11 KB for :

1. First-fit 2. Best-fit 3. Worst-fit. Draw the solution for each one.

Q3: A user program consists of 26 instructions, each instruction takes one byte. Suppose page size = frame size = 4 bytes. The physical memory size is 512 bytes.

a. Calculate the number of pages in the logical map.

b. Calculate the number of frames in the physical map.

c. Draw the logical map.

d. Suppose page 0 is allocated in frame 10, and page 3 is allocated in frame 30, calculate the physical addresses of the logical addresses of page 0 and page 3.

e. Draw the physical map.

Q4: A user program consists of 16 instructions, each instruction takes two bytes. Suppose page size = frame size = 8 bytes. The physical memory size is 512 bytes.

a. Calculate the number of pages in the logical map.
b. Calculate the number of frames in the physical map.
c. Draw the logical map.
d. Suppose page 0 is allocated in frame 10, and page 3 is allocated in frame 30, calculate the physical addresses of the logical addresses of page 0 and page 3.
e. Draw the physical map.

Q5: A user program of size 35 bytes. Suppose page size = frame size = 8 bytes. The physical memory size is 8192 bytes.

a. Calculate the number of pages in the logical map.
b. Calculate the number of frames in the physical map.
c. Draw the logical map.
d. Suppose page 1 is allocated in frame 10, and page 3 is allocated in frame 30, calculate the physical addresses of the logical addresses of page 1 and page 3.
e. Draw the physical map.

Q6: Given free memory partitions of
100k, 500k, 200k, 700k, and 600k (in order),
How would each of the First-fit, Best-fit, and Worst-fit algorithms place processes of 212k, 417k, 112k, and 426k (in order)?
Which algorithm makes the most efficient use of memory?

Q7: A process uses six pages numbered (0, 1, 2, 3, 4, 5) during its execution. The pages 0, 2, 4, and 5 are loaded in main memory at page frames 4, 2, 5, and 1 respectively. Assume the page size is 4 bytes and the main memory of size 32 bytes is used to map the pages.

a. Construct the page map table.
b. Find the physical addresses for the following logical addresses.
(0,2), (1,3), (2,0), (4,1), (3,3), and (5,2)

Q8: Consider the following part of process segment table:

Segment number	Limit	Base address
0	1100	1200
1	500	4400
2	700	3500
3	950	2200
4	1200	7500

What are the physical addresses for the following logical addresses?
a. (0.1050) b. (1.550) c. (2.770) d. (3.900) e. (4.1100)

Q9: Consider a system in which memory consists of the following hole sizes in the following order:
20kb, 4kb, 25kb, 9kb, 12kb, 7kb, 10kb, and 28kb.
Which hole is taken for successive block request of
22kb, 10kb, 8kb.
For First-fit, Best-fit, and Worst-fit?

Q10: A user program consists of 30 instructions, each instruction takes one byte. Suppose page size = frame size = 9 bytes. The physical memory size is 512 bytes. (2015)

a. Calculate the number of pages in the logical map.
b. Calculate the number of frames in the physical map.
c. Draw the logical map.
d. Suppose page 0 is allocated in frame 10, and page 2 is allocated in frame 15, calculate the physical addresses of the logical addresses of page 0 and page 2.
e. Draw the physical map.

Q11: Given memory partitions as shown below. How would each of the First fit, Worst fit, and Best fit algorithms place processes (jobs) of 5k, 10 k, and 20k(in order), in the holes (partitions) of the memory?

Used 10k	Hole 10k	Used 20k	Hole 30k	Used 10k	Hole 5k	Used 60k	Hole 15k	Used 20k	Hole 20k