BAHRIA UNIVERSITY LAHORE CAMPUS

ASSIGNMENT

SUBJECT: Operating System

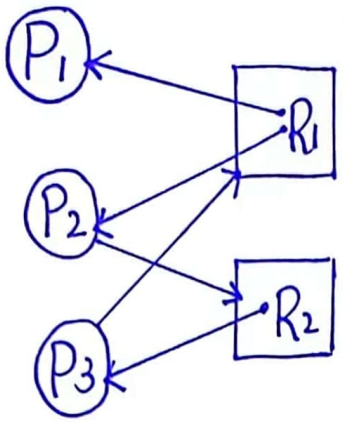
Submitted To: Muhammad Mudassar

Submitted By: Mehak

Enrollment No: 03-134221-015

Semester: BSCS 5A

Q.1 Resource Allocation Graph



Process	Allocate R1	Allocate R2	Request R1	Request R2
P1	1	0	0	0
P2	1	0	0	1
P3	0	1	1	0

Available resource = 0 0

P1 Now = 1 0

new = 1 0

P3 = 0 1

new = 1 1

P_2 = 1 0

2 1

a) This graph contain no deadlock because all are terminated successfully and There is no cycle. possible execution sequence is p1, p3, p2.

b) if P1 demand R_2 that will put system into deadlock none of processes will get all resource and circular wait will occur there.

c) Pu request for R1 cannot be fulfilled as it would introduce another process waiting for resource held by another process there will no deadlock only possible equation is p1, p3, p2.

Q-2.

Processes	Allocation				Max Need				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	0	1	0	0	7	5	3	0	8	10	8	6
P_1	2	0	0	0	3	2	2	0				
P_2	3	0	2	0	9	0	2	0				
P_3	2	1	1	0	2	2	2	0				
P_4	0	0	2	2	4	3	3	2				

Need = Max - Allocation

Need			
A	B	C	D
7	4	3	0
1	2	2	0
6	0	0	0
0	1	1	0
4	3	1	0

Finish

0	~~F~~	T
1	~~F~~	T
2	~~F~~	T
3	~~F~~	T
4	~~F~~	T

safe sequence

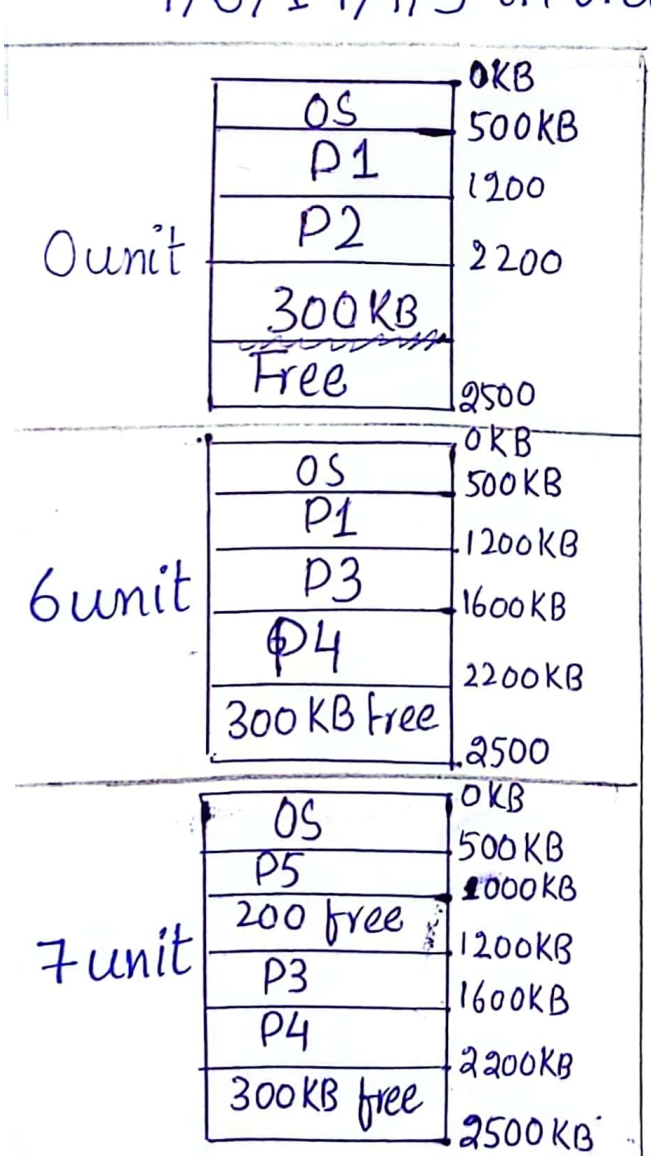
<p0, p1, p2, p3, p4>

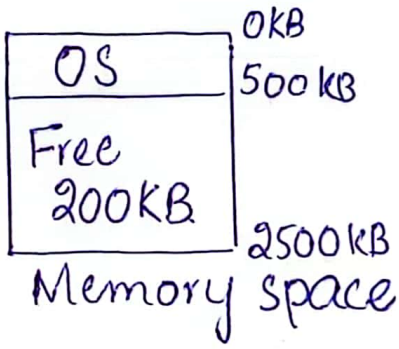
work

8 10 8 6
+ 0 1 0 0
8 11 8 6
+ 2 0 0 0
10 11 8 6
+ 3 0 2 0
13 11 10 6
+ 2 1 1 0
15 12 11 6
+ 0 0 2 2
15 12 13 8

b) Suppose p1 request (1,0,1,1) resource so it will be granted immediately as available resource is greater than (1,0,1,1) but if it request (10,9,6,2) so this will not be granted that is smaller than available one.

Q.3 memory partition of size 2500 KB in which first 0 KB to 500KB is allocated to OS
Rest there are 5 process requiring sizes 700, 1000, 400, 600, 500 and execution time is 7, 6, 14, 9, 3 in order





Q.4 5 partition memory of fixed sizes as 200KB, 400KB, 350KB, 250KB, 500KB and 5 processes that require memory as 180KB, 220KB, 320KB, 190KB, 150KB apply first fit and worst fit.

First Fit

Memory	200KB	400KB	350KB	250KB	500KB
Processes					
180KB	20KB				
220KB		180KB			
320KB			30KB		
190KB				60KB	
150KB		30KB			

Worst Fit

Memory	200KB	400KB	350KB	250KB	500KB
180KB					320KB
220KB		180KB			
320KB			30KB		
190KB					130KB
150KB				100KB	

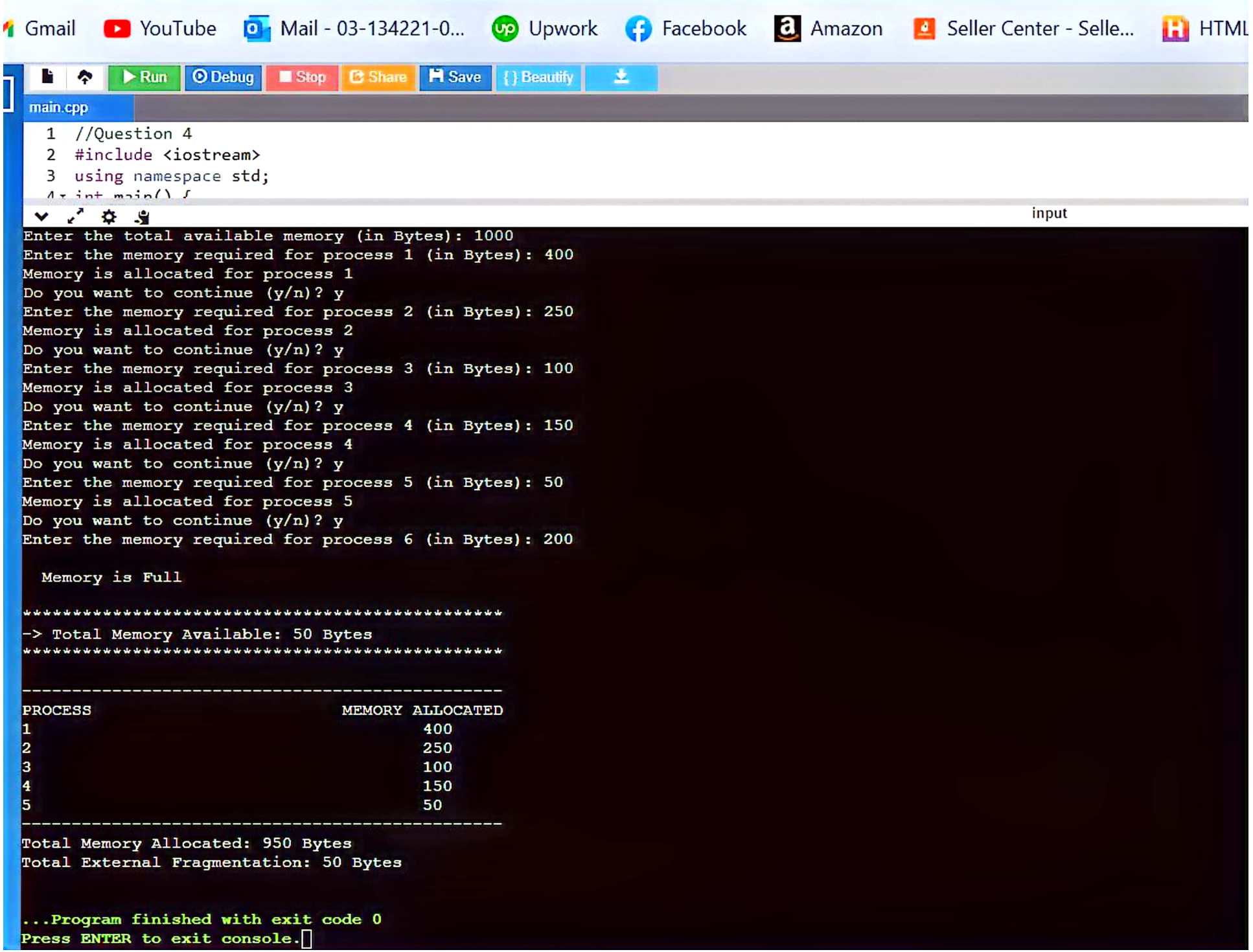
In worst case there is only 200KB memory which has been not assigned to any process and

There is no common memory partition that is not allocated to any process in both strategies.

main.cpp

```cpp
//Question 4
#include <iostream>
using namespace std;
int main() {
    const int MinMemory= 1;
    int totalMemory;
        int allocatedMemories[100];
    int process = 0;
    int totalAllocatedMemory = 0;
 cout << "Enter the total available memory (in Bytes): ";
 cin >> totalMemory;

    do {
        int memoryRequired;
        cout << "Enter the memory required for process " << process + 1 << " (in Bytes): ";
      cin >> memoryRequired;

        if (memoryRequired < MinMemory) {
            cout << "Memory required must be at least " << MinMemory << " Bytes." << endl;
            continue;
        }

        if (memoryRequired > totalMemory - totalAllocatedMemory) {
       cout<<endl;
            cout << "  Memory is Full " << endl;

            break;
        }

       cout << "Memory is allocated for process " << process + 1 << endl;
        allocatedMemories[process] = memoryRequired;
        totalAllocatedMemory += memoryRequired;
        process++;

        char choice;
       cout << "Do you want to continue (y/n)? ";
        cin >> choice;
        if (choice == 'n') {
            break;
        }
    } while (true);
```

```
        }
    } while (true);

    int availableMemory = totalMemory - totalAllocatedMemory;
    cout<<endl;
   cout<<"************************************************"<<endl;
   cout << "-> Total Memory Available: " << availableMemory << " Bytes" << endl;
    cout<<"************************************************"<<endl;
    cout<<endl;
   cout<<"------------------------------------------------"<<endl;
   cout << "PROCESS\t\t\t\tMEMORY ALLOCATED" << endl;
    for (int i = 0; i < process; i++) {
        cout << i + 1 << "\t\t\t\t\t" << allocatedMemories[i] << endl;
    }
cout<<"------------------------------------------------"<<endl;
    cout << "Total Memory Allocated: " << totalAllocatedMemory << " Bytes" << endl;
   cout << "Total External Fragmentation: " << availableMemory << " Bytes" << endl;

    return 0;
}
```


Gmail
YouTube
Mail - 03-134221-0...
Upwork
Facebook
Amazon
Seller Center - Selle...
HTML
Run
Debug
Stop
Share
Save
{ } Beautify
main.cpp
1 //Question 4
2 #include <iostream>
3 using namespace std;
input
Enter the total available memory (in Bytes): 1000
Enter the memory required for process 1 (in Bytes): 400
Memory is allocated for process 1
Do you want to continue (y/n)? y
Enter the memory required for process 2 (in Bytes): 250
Memory is allocated for process 2
Do you want to continue (y/n)? y
Enter the memory required for process 3 (in Bytes): 100
Memory is allocated for process 3
Do you want to continue (y/n)? y
Enter the memory required for process 4 (in Bytes): 150
Memory is allocated for process 4
Do you want to continue (y/n)? y
Enter the memory required for process 5 (in Bytes): 50
Memory is allocated for process 5
Do you want to continue (y/n)? y
Enter the memory required for process 6 (in Bytes): 200
Memory is Full
-> Total Memory Available: 50 Bytes
PROCESS MEMORY ALLOCATED
1 400
2 250
3 100
4 150
5 50
Total Memory Allocated: 950 Bytes
Total External Fragmentation: 50 Bytes
...Program finished with exit code 0
Press ENTER to exit console.