Operating Systems

Experiment 5

Ayush Jain|60004200132|SE-B2

Aim: Write a code for First fit, Best fit, Worst fit, Next fit.

Code:

1. **First fit:**

```
#include <stdio.h>
int main()
{

int totalMem = 0;
int part[] = {200, 400, 600, 500, 300, 250}; int
i, j;
printf("\nEnter number of process to be added to main memory:"); int
n ;
scanf("%d", &n);
int mem_p[n];
int flag[6];
for(i = 0; i < n; i++)
{
printf("Enter memory to be assigned to process %d :",(i+1));
scanf("%d", &mem_p[i]);
}
for(i = 0; i < 6; i++)
flag[i] = 0; int id;
for(i = 0; i < n; i++)
{ id = -
1;
for(j = 0; j < 6; j++)
{
if((flag[j] == 0) && (mem_p[i] <= part[j]))
{ id =
j;
break;
}
} if(id != -
1)
```

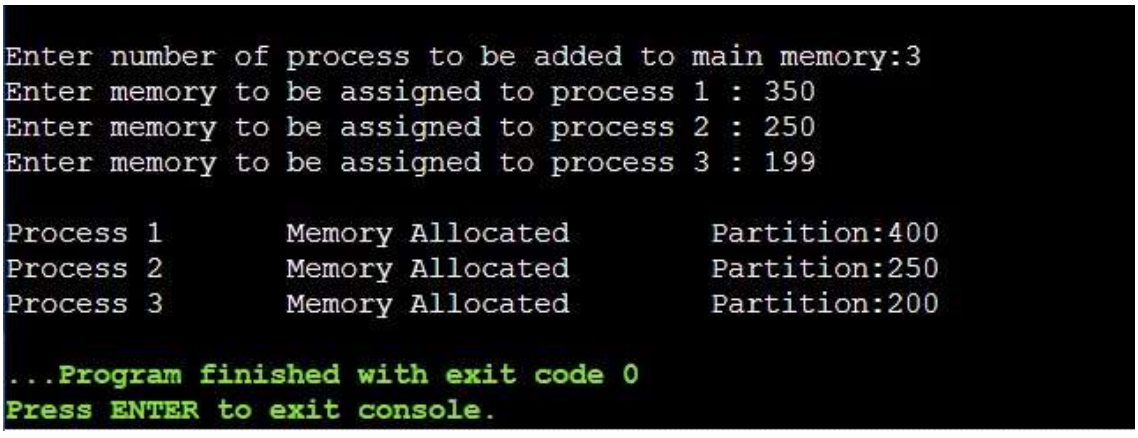
```
{
printf("\nProcess %d\tMemory Allocated\tPartition:%d ",(i+1),part[id]);
flag[id] = 1;
}
else
printf("\nProcess %d\tMemory Not Allocated", (i+1));
}
 return 0;
}
```

```
Enter number of process to be added to main memory:3
Enter memory to be assigned to process 1 :500
Enter memory to be assigned to process 2 :30
Enter memory to be assigned to process 3 :
700

Process 1       Memory Allocated        Partition:600
Process 2       Memory Allocated        Partition:200
Process 3       Memory Not Allocated

...Program finished with exit code 0
Press ENTER to exit console.
```

2. Best fit:

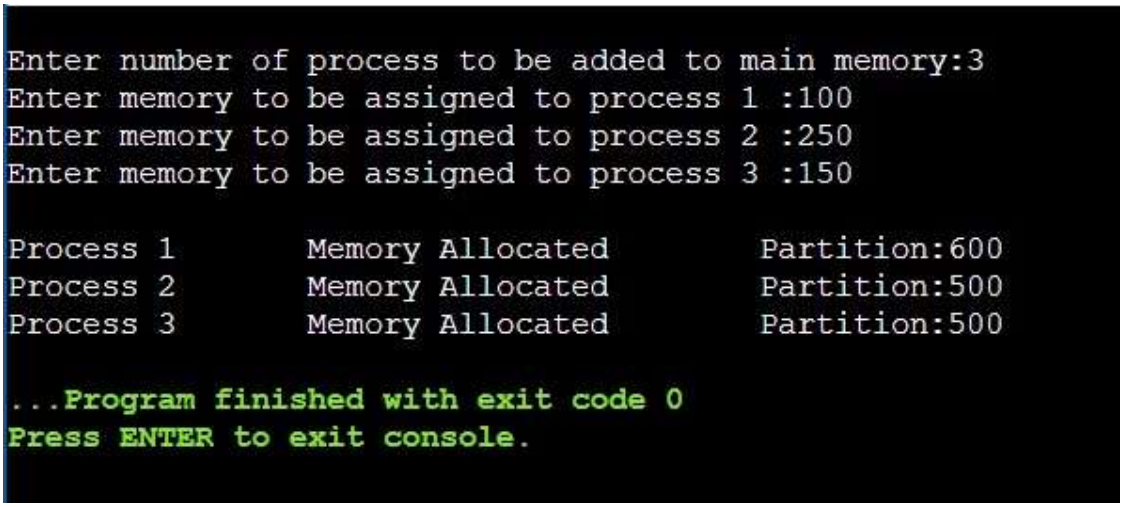
```
#include <stdio.h>
int main()
{
int totalMem = 0;
int part[] = {200, 400, 600, 500, 300, 250}; int
i, j;
printf("\nEnter number of process to be added to main memory:"); int
n ;
scanf("%d", &n); int
mem_p[n]; int
flag[6]; for(i = 0; i <
n; i++)
{
printf("Enter memory to be assigned to process %d : ",(i+1));
scanf("%d", &mem_p[i]);
}
```

```
for(i = 0; i < 6; i++)
flag[i] = 0;  int diff =
10000, id;
for(i = 0; i < n; i++)
{ id = -
1;
for(j = 0; j < 6; j++)
{
if((flag[j] == 0) && (mem_p[i] <= part[j]) && (part[j] - mem_p[i]
< diff))
{
diff = part[j] - mem_p[i]; id
= j;
}
}
if(id != -1)
{
printf("\nProcess %d\tMemory Allocated\tPartition:%d ",(i+1),part[id]);
flag[id] = 1;
}
else
 printf("\nProcess %d\tMemory Not Allocated", (i+1));
diff = 10000;
}
 return 0;
}
```



3. Worst fit

```
#include <stdio.h>
int main()
{
int totalMem = 0;
int part[] = {200, 400, 600, 500, 300, 250}; int
i, j;
printf("\nEnter number of process to be added to main memory:"); int
n ;
scanf("%d", &n); int
mem_p[n];
for(i = 0; i < n; i++)
{
printf("Enter memory to be assigned to process %d :",(i+1));
scanf("%d", &mem_p[i]);
} int diff = 0,
id;
for(i = 0; i < n; i++)
{ id = -
1;
for(j = 0; j < 6; j++)
{
   if((mem_p[i] <= part[j]) && (part[j] - mem_p[i] > diff))
{
diff = part[j] - mem_p[i]; id
= j;
}
}  if(id != -
1)
{
printf("\nProcess %d\tMemory Allocated\tPartition:%d ",(i+1),part[id]);
part[id] = part[id] - mem_p[i];
}
else
 printf("\nProcess %d\tMemory Not Allocated", (i+1));
diff = 0;
 }
 return 0;
}
```



4. Next fit

```
#include <stdio.h>
int main()
{

int part[] = {200, 400, 600, 500, 300, 250}; int
i, j;
printf("\nEnter number of process to be added to main memory:"); int
n ;
scanf("%d", &n); int
mem_p[n];
int flag[6]; for(i =
0; i < n; i++)
{
printf("Enter memory to be assigned to process %d : ",(i+1));
scanf("%d", &mem_p[i]);
}
for(i = 0; i < 6; i++)
flag[i] = 0; int id,
prevId = 0;
for(i = 0; i < n; i++)
{ id = -
1;
for(j = prevId; j < 6; j++)
{
if((flag[j] == 0) && (mem_p[i] <= part[j]))
{ id =
j;
```

```
brea
k;
}
}
 if(id != -1)
{
printf("\nProcess %d\tMemory Allocated\tPartition:%d ",(i+1),part[id]);
flag[id] = 1; prevId = id;
}
else
 printf("\nProcess %d\tMemory Not Allocated", (i+1));
 }
 return 0;
}
```

```
Enter number of process to be added to main memory:3
Enter memory to be assigned to process 1 : 250
Enter memory to be assigned to process 2 : 100
Enter memory to be assigned to process 3 : 350

Process 1       Memory Allocated        Partition:400
Process 2       Memory Allocated        Partition:600
Process 3       Memory Allocated        Partition:500

...Program finished with exit code 0
Press ENTER to exit console.
```

Conclusion:

Learning and executing code for First fit, Best fit, Worst fit, Next fit.