Ex. No.: 10b)
Date: 11/4/2025

FIRST FIT

Aim:

To write a C program for implementation memory allocation methods for fixed partition using first fit.

Algorithm:

1. Define the max as 25.
2: Declare the variable frag[max],b[max],f[max],i,j,nb,nf,temp, highest=0, bf[max],ff[max]. 3: Get the number of blocks,files,size of the blocks using for loop.
4: In for loop check bf[j]!=1, if so temp=b[j]-f[i]
5: Check highest

Program Code:

```
#include<stdio.h>
int main(){
    int n, m;
    scanf("%d", &n);
    scanf("%d", &m);
    int block[n];
    int process[m];
    int allocation[m];
    for(int i=0; i<m; i++){
        allocation[i]=-1;
    }
    int occupied[n];
    for(int i=0; i<n; i++)
    {
    occupied[i]=0;
    }
}
```

```
for(int i=0;i<n;i++){
    scanf("%d",&block[i]);
}
for(int i=0;i<m;i++){
    scanf("%d",&process[i]);
}
for(int i=0;i<m;i++)
{
for(int j=0;j<n;j++)
{
if(!occupied[j]&&blocks[j]>=process[i])
{
allocation[i]=j;
occupied[j]=1;
blocks[j]=process[i];
Printf("%d",block[j]);
break;
}
}
}
Printf("\n process NO Process size block NO");
for(int i=0;i<m;i++)
{
if(allocation[i]!=-1)
{
Printf("\n %d\t\t %d\t\t %d",i+1,process[i],
allocation[i]+1);
}else
Printf("\n %d\t\t %d\t\t Not allocation",i++
Process[i]);
}
}
}
```

Output

Enter no of blocks : 4

Block size :

B1 - 100

B2 - 500

B3 - 150

B4 - 300

Enter no of processes : 3

Process size :

P_1 - 99

P_2 - 211

P_3 - 300

Process NO	Process size	Block NO
P_1	99	B1
P_2	211	B2
P_3	300	B4

Sample Output:

```
Enter the number of blocks:4
Enter the number of files:3

Enter the size of the blocks:-
Block 1:5
Block 2:8
Block 3:4
Block 4:10
Enter the size of the files:-
File 1:1
File 2:4
File 3:7

File_no:        File_size :     Block_no:       Block_size:     Fragment
1               1               1               5               4
2               4               2               8               4
3               7               4               10              3_
```

Result: Hence first fit memory management was successfully executed