S.No: 10	Exp. Name: ***Implementation of worst fit allocation technique.***	Date:2025-04-12

ID: 2300331530046

Aim:

Aim: Implementation of worst fit allocation technique.

Description: One of the simplest methods for memory allocation is to divide memory into several fixed-sized partitions. Each partition may contain exactly one process. In this multiple- partition method, when a partition is free, a process is selected from the input queue and is loaded into the free partition. When the process terminates, the partition becomes available for another process. The operating system keeps a table indicating which parts of memory are available and which are occupied. Finally, when a process arrives and needs memory Worst-fit chooses the largest available block.

Source Code:

worstfitalloc.c

```
#include<stdio.h>
#include<limits.h>
#define max 25
void main()
{
   int flag[max],b[max],f[max],i,j,nb,nf,temp;
   static int bf[max],ff[max];
   printf("\tMemory Management Scheme - First Fit");
   printf("\nEnter the number of blocks:");
   scanf("%d",&nb);
   printf("Enter the number of files:");
   scanf("%d",&nf);
   printf("Enter the size of the blocks:-\n");
   for(i=1;i<=nb;i++)
      {
         printf("Block %d:",i);
         scanf("%d",&b[i]);
      }
   printf("Enter the size of the files :-\n");
   for(i=1;i<=nf;i++)
      {
         printf("File %d:",i);
         scanf("%d",&f[i]);
      }
   for(i=1;i<=nf;i++)
      {
         for(j=1;j<=nb;j++)
            {
               if(bf[j]!=1)
               {
                  temp=b[j]-f[i];
                  if(temp>0)
                  {
                     ff[i]=j;
                     break;
                  }
               }
            }
         flag[i]=temp;
```

```
            bf[ff[i]]=1;
        }
    printf("File_no:\tFile_size :\tBlock_no:\tBlock_size:\tFragement");
    for(i=1;i<=nf;i++)
        printf("\n%d\t\t%d\t\t%d\t\t%d\t\t%d",i,f[i],ff[i],b[ff[i]],flag[i]);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1				
User Output				
Memory Management Scheme - First Fit 3				
Enter the number of blocks: 3				
Enter the number of files: 2				
Enter the size of the blocks:- 5				
Block 1: 5				
Block 2: 2				
Block 3: 7				
Enter the size of the files :- 1				
File 1: 1				
File 2: 4				
File_no:	File_size :	Block_no:	Block_size:	Fragement
1	1	1	5	4
2	4	3	7	3

Test Case - 2				
User Output				
Memory Management Scheme - First Fit 2				
Enter the number of blocks: 2				
Enter the number of files: 2				
Enter the size of the blocks:- 22				
Block 1: 22				
Block 2: 23				
Enter the size of the files :- 2				
File 1: 2				
File 2: 4				
File_no:	File_size :	Block_no:	Block_size:	Fragement
1	2	1	22	20
2	4	2	23	19