**13. Construct a C program to implement various memory allocationstrategies**

#include<stdio.h>

void bestfit(int mp[],int p[],int m,int n){int

j=0;

for(int i=0;i<n;i++){

if(mp[i]>p[j]){

printf("\n%d fits in %d",p[j],mp[i]);

mp[i]=mp[i]-p[j++];

i=i-1;

}

}

for(int i=j;i<m;i++)

{

printf("\n%d must wait for its process",p[i]);

}

}

void rsort(int a[],int n){ for(int

i=0;i<n;i++){

for(int j=0;j<n;j++){

if(a[i]>a[j]){

int t=a[i];

a[i]=a[j];

a[j]=t;

}

}

}}

void sort(int a[],int n){ for(int

i=0;i<n;i++){

for(int j=0;j<n;j++){

if(a[i]<a[j]){

int t=a[i];

a[i]=a[j];

a[j]=t;

}

}

}

}

void firstfit(int mp[],int p[],int m,int n){

sort(mp,n);

sort(p,m);

bestfit(mp,p,m,n);

}

void worstfit(int mp[],int p[],int m,int n){

rsort(mp,n);

sort(p,m);

bestfit(mp,p,m,n);

}

int main(){

int m,n,mp[20],p[20],ch; printf("Number of

memory partition : ");scanf("%d",&n);

printf("Number of process : ");

scanf("%d",&m);

printf("Enter the memory partitions : \n");for(int

i=0;i<n;i++){

scanf("%d",&mp[i]);

}

printf("ENter process size : \n");

for(int i=0;i<m;i++){scanf("%d",&p[i]);

}

printf("1. Firstfit\t2. Bestfit\t3. worstfit\nEnter your choice : ");

scanf("%d",&ch);

switch(ch){

case 1:

bestfit(mp,p,m,n);

break;

case 2:

firstfit(mp,p,m,n);

break;

case 3:

worstfit(mp,p,m,n);

break;

default:

printf("invalid");

break;

}

}

