# EXPERIMENT NO: 3(a)

## Aim: Simulate Multiprogramming with a fixed number of tasks(MFT)

PROGRAM:

#include<stdio.h>  
int main()  
{  
int m,p,s,p1;  
int m1[4],i,f,f1=0,f2=0,fra1,fra2,s1,pos;  
printf("Enter the memory size:");  
scanf("%d",&m);  
printf("Enter the no of partitions:");  
scanf("%d",&p);  
s=m/p;  
printf("Each partn size is:%d",s);  
printf("\nEnter the no of processes:");  
scanf("%d",&p1);  
pos=m;  
for(i=0;i<p1;i++)  
{  
if(pos<s)  
{  
printf("\nThere is no further memory for process%d",i+1);  
m1[i]=0;  
break;  
}  
else  
{  
printf("\nEnter the memory req for process%d:",i+1);  
scanf("%d",&m1[i]);  
if(m1[i]<=s)  
{  
printf("\nProcess is allocated in partition%d",i+1);  
fra1=s-m1[i];  
printf("\nInternal fragmentation for process is:%d",fra1);  
f1=f1+fra1;  
pos=pos-s;  
}  
else  
{  
printf("\nProcess not allocated in partition%d",i+1);  
s1=m1[i];  
while(s1>s)  
{  
s1=s1-s;  
pos=pos-s;  
}  
pos=pos-s;  
fra2=s-s1;  
f2=f2+fra2;  
printf("\nExternal Fragmentation for this process is:%d",fra2);  
}  
}  
}  
printf("\nProcess\tallocatedmemory");  
for(i=0;i<p1;i++)  
printf("\n%5d\t%5d",i+1,m1[i]);  
f=f1+f2;  
printf("\nThe tot no of fragmentation is:%d",f);  
return 0;  
}

## EXPERIMENT NO: 3(b)

AIM:Simulate Multiprogramming with a variable number of tasks

# PROGRAM:

#include<stdio.h>  
void main()  
{  
int m=0,m1=0,m2=0,p,count=0,i;  
printf("enter the memory capacity:");  
scanf("%d",&m);  
printf("enter the no of processes:");  
scanf("%d",&p);  
for(i=0;i<p;i++)  
{  
printf("\nenter memory req for process%d: ",i+1);  
scanf("%d",&m1);  
count=count+m1;  
if(m1<=m)  
{  
if(count==m)  
printf("there is no further memory remaining:");  
printf("the memory allocated for process%d is: %d ",i+1,m);  
m2=m-m1;  
printf("\nremaining memory is: %d",m2);  
m=m2;  
}  
else  
{  
printf("memory is not allocated for process%d",i+1);  
}  
printf("\nexternal fragmentation for this process is:%d",m2);  
}  
}