**PRACTICAL-3**

**IMPLEMENTATION OF WORST,BEST AND FIRST FIT FOR CONTIGUOUS ALLOCATION ASSUMING RANDOMLY GENRATED FREE SPACE LIST**

#include<stdio.h>

#include<conio.h>

void main()

{

int store[75],i,n,t=0,count=0,j,a[' '],ch,min,flag=0;

char choice='y';

clrscr();

for(i=0;i<75;++i)

{

if(i<10)

store[i]=0;

if(i>=10&&i<25)

store[i]=1;

if(i>=25&&i<50)

store[i]=0;

if(i>=50&&i<71)

store[i]=1;

}

printf("free block description:\n");

printf("block 1: 0-9\nblock 2: 26-49\nblock 3: 71-74\n");

while(choice=='y')

{

printf("\nenter the no. of blocks you want to allocate:");

scanf("%d",&n);

printf("enter the strategy you want to use:");

printf("\n1.first fit\n2.best fit\n3.worst fit\n");

printf("enter the choice:");

scanf("%d",&ch);

switch(ch)

{

case 1 : for(i=0;i<75;++i)

{

if(store[i]==0)

{

flag=1;

count++;

if(count>=n)

{

printf("allocated from %d",i-n+1);

break; }

}

else

{

count=0;

if(flag==1)

++j;

flag=0;

}

}

break;

case 2 : t=0;

count=0;

j=0;

for(i=0;i<75;++i)

{

if(store[i]==0)

{

flag=1;

++count;

//if(count>=n)

a[j]=count;

}

else

{

count=0;

if(flag==1)

{

j=j+1;

flag=0;

}

}

}

min=a[0];

for(i=1;i<j-1;++i)

if((min>a[i])&&(a[i]>=n))

t=i;

printf("\nallocated from %d",t);

break;

case 3 : t=0;

count=0;

j=0;

for(i=0;i<75;++i)

{

if(store[i]==0)

{

flag=1;

count++;

a[j]=count;

}

else

{

count=0;

if(flag==1)

{

++j;

flag=0;

}

}

}

min=a[0];

for(i=1;i<j;++i)

if((min<a[i])&&(a[i]>=n))

t=i;

printf("\nmemory allocated in block %d",t);

}

printf("\nwant to repeat for other strategy(y/n)");

scanf(" %c",&choice);

}

}