#include<iostream>

#include<stdlib.h>

using namespace std;

int main()

{

int NOP,NOB,i,j,ch;

cout<<"Enter The number of processes: ";

cin>>NOP;

cout<<"Enter the number of blocks: ";

cin>>NOB;

int processes[NOP],blocks[NOB],c[NOB],d[NOB],flag[NOB],alloc[NOB];

// NOP = NOB

cout<<"Enter Processes size: \n";

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

{

cout<<"P["<<i<<"]: ";

cin>>processes[i];

}

cout<<"\nEnter Blocks size: \n";

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

{

cout<<"B["<<i<<"]: ";

cin>>blocks[i];

c[i]=blocks[i];d[i]=blocks[i];

}

if(NOP<=NOB)

{

do

{

cout<<"\n\n1.First fit\t2.Best fit \t3.Worst fit\t4.Exit";

cout<<"\nEnter your choice : ";

cin>>ch;

switch(ch)

{

case 1: cout<<"\nFirst Fit\n";

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

{

for(j=0;j<NOB;j++)

{

if(processes[i]<=blocks[j])

{

alloc[j]=processes[i];cout<<"\nAlloc["<<alloc[j]<<"]";

cout<<"\nProcess "<<i<<":"<<processes[i]<<" Block: "<<j<<": "<<blocks[j];

flag[i]=0,blocks[j]=0;break;

}

else flag[i]=1;

}

}

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

{

if(flag[i]!=0)

cout<<"\nProcess "<<i<<" of size "<<processes[i]<<" is not allocated\n";

}break;

case 2: cout<<"\nBest Fit\n";

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

{

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

{

if(c[i]>c[j])

{

int temp=c[i];c[i]=c[j];c[j]=temp;

}

}

}

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

{

for(j=0;j<NOB;j++)

{

if(processes[i]<=c[j])

{

alloc[j]=processes[i];cout<<"\nAlloc["<<alloc[j]<<"]";

cout<<"\nProcess "<<i<<":"<<processes[i]<<" Block: "<<j<<": "<<c[j];

flag[i]=0,c[j]=0;break;

}else flag[i]=1;

}

}

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

{

if(flag[i]!=0)

cout<<"\nProcess "<<i<<" of size "<<processes[i]<<" is not allocated\n";

}

break;

case 3: cout<<"\nWorst Fit\n";

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

{

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

{

if(d[i]<d[j])

{

int temp=d[i];d[i]=d[j];d[j]=temp;

}

}

}

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

{

for(j=0;j<NOB;j++)

{

if(processes[i]<=d[j])

{

alloc[j]=processes[i];cout<<"\nAlloc["<<alloc[j]<<"]";

cout<<"\nProcess "<<i<<":"<<processes[i]<<" Block: "<<j<<": "<<d[j];

flag[i]=0,d[j]=0;break;

}else flag[i]=1;

}

}

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

{

if(flag[i]!=0)

cout<<"\nProcess "<<i<<" of size "<<processes[i]<<" is not allocated\n";

}

break;

case 4: exit(0);

default: cout<<"Invalid Choice…!";break;

}

}while(ch<=3);

}else cout<<"\nNumber of Process must be less then or equal to number of blocks....";

return 0;

}

**Output:**

Enter The number of processes: 4

Enter the number of blocks: 5

Enter Processes size:

P[0]: 212

P[1]: 417

P[2]: 112

P[3]: 426

Enter Blocks size:

B[0]: 100

B[1]: 500

B[2]: 200

B[3]: 300

B[4]: 600

1.First fit 2.Best fit 3.Worst fit 4.Exit

Enter your choice : 1

First Fit

Alloc[212]

Process 0:212 Block: 1: 500

Alloc[417]

Process 1:417 Block: 4: 600

Alloc[112]

Process 2:112 Block: 2: 200

Process 3 of size 426 is not allocated

1.First fit 2.Best fit 3.Worst fit 4.Exit

Enter your choice : 2

Best Fit

Alloc[212]

Process 0:212 Block: 2: 300

Alloc[417]

Process 1:417 Block: 3: 500

Alloc[112]

Process 2:112 Block: 1: 200

Alloc[426]

Process 3:426 Block: 4: 600

1.First fit 2.Best fit 3.Worst fit 4.Exit

Enter your choice : 3

Worst Fit

Alloc[212]

Process 0:212 Block: 0: 600

Alloc[417]

Process 1:417 Block: 1: 500

Alloc[112]

Process 2:112 Block: 2: 300

Process 3 of size 426 is not allocated

1.First fit 2.Best fit 3.Worst fit 4.Exit

Enter your choice : 4

Process returned 0 (0x0) execution time : 13.287 s

Press any key to continue.