**First Fit**

**Program**

#include<stdio.h>

void main(){

int memblok,limit,i,j;

printf("Enter the number of memory blocks and no of process\n");

scanf("%d%d",&memblok,&limit);

struct memory{

int size,alloc;

}m[memblok];

struct process{

int psize,flag;

}p[limit];

printf("Enter the size of memory block\n");

for(i=0;i<memblok;i++){

scanf("%d",&m[i].size);

m[i].alloc=0;

}

printf("Enter the size of processes\n");

for(i=0;i<limit;i++){

scanf("%d",&p[i].psize);

p[i].flag=0;

}

for(i=0;i<limit;i++){

for(j=0;j<memblok;j++){

if(p[i].flag==0){

if(p[i].psize<=m[j].size){

if(m[j].alloc==1)

continue;

else{

m[j].alloc=1;

p[i].flag=1;

printf("process %d is allocated in %d block\n",p[i].psize,m[j].size);

}

}

}

}

}

for(i=0;i<limit;i++){

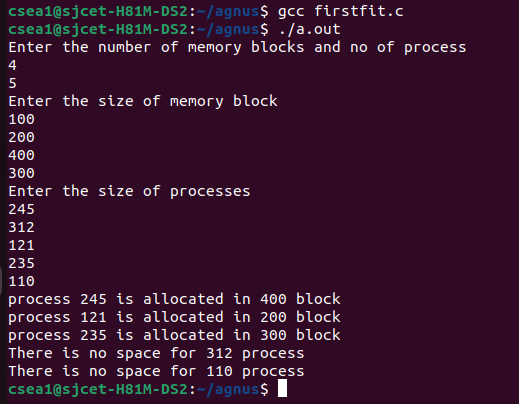
if(p[i].flag==0)

printf("There is no space for %d process\n",p[i].psize);

}

}

**Output**

****

**Best Fit**

**Program**

#include<stdio.h>

void main(){

int memblok,limit,i,j;

printf("Enter the number of memory blocks and no of process\n");

scanf("%d%d",&memblok,&limit);

struct memory{

int size,alloc;

}m[memblok];

struct process{

int psize,flag;

}p[limit];

printf("Enter the size of memory block\n");

for(i=0;i<memblok;i++){

scanf("%d",&m[i].size);

m[i].alloc=0;

}

for(i=0;i<memblok;i++){

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

if(m[i].size>=m[j].size){

int temp=m[i].size;

m[i].size=m[j].size;

m[j].size=temp;

}

}

}

printf("Enter the size of processes\n");

for(i=0;i<limit;i++){

scanf("%d",&p[i].psize);

p[i].flag=0;

}

for(i=0;i<limit;i++){

for(j=0;j<memblok;j++){

if(p[i].flag==0){

if(p[i].psize<=m[j].size){

if(m[j].alloc==1)

continue;

else{

m[j].alloc=1;

p[i].flag=1;

printf("process %d is allocated in %d block\n",p[i].psize,m[j].size);

}

}

}

}

}

for(i=0;i<limit;i++){

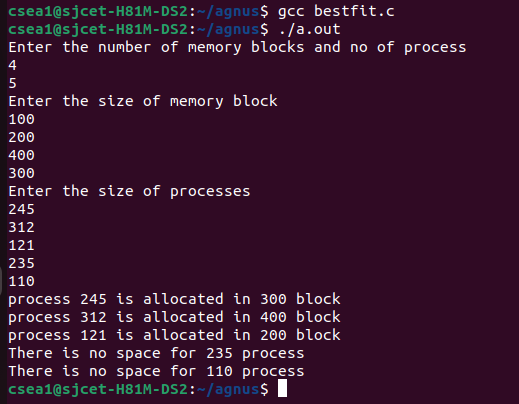
if(p[i].flag==0)

printf("There is no space for %d process\n",p[i].psize);

}

}

**Output**

****

**Worst Fit**

**Program**

#include<stdio.h>

void main(){

int memblok,limit,i,j;

printf("Enter the number of memory blocks and no of process\n");

scanf("%d%d",&memblok,&limit);

struct memory{

int size,alloc;

}m[memblok];

struct process{

int psize,flag;

}p[limit];

printf("Enter the size of memory block\n");

for(i=0;i<memblok;i++){

scanf("%d",&m[i].size);

m[i].alloc=0;

}

for(i=0;i<memblok;i++){

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

if(m[i].size<=m[j].size){

int temp=m[i].size;

m[i].size=m[j].size;

m[j].size=temp;

}

}

}

printf("Enter the size of processes\n");

for(i=0;i<limit;i++){

scanf("%d",&p[i].psize);

p[i].flag=0;

}

for(i=0;i<limit;i++){

for(j=0;j<memblok;j++){

if(p[i].flag==0){

if(p[i].psize<=m[j].size){

if(m[j].alloc==1)

continue;

else{

m[j].alloc=1;

p[i].flag=1;

printf("process %d is allocated in %d block\n",p[i].psize,m[j].size);

}

}

}

}

}

for(i=0;i<limit;i++){

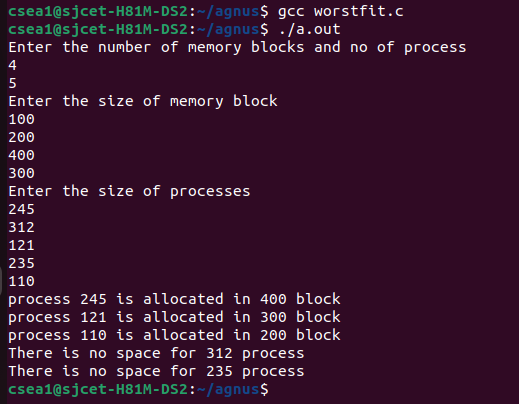
if(p[i].flag==0)

printf("There is no space for %d process\n",p[i].psize);

}

}

**Output**

****