



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

CSE2005: OPERATING SYSTEMS

Fall Semester 20-21: LAB FAT

GROUP 3

ANUSHKA DIXIT [19BCE0577]

Question: Allocate/free memory to processes in whole pages, find max allocatable pages, incorporate address translation into the program.

Code:

```
//Q3) Allocate/free memory to processes in whole pages,
//find max allocatable pages, incorporate address translation into the program
.

#include<stdio.h>
#include<stdlib.h>
#include<time.h>

struct MainMem
{
    int start_add;
    int pn;
};

int main()
{
    int s,i,totPages,pno,offset,n,r,arr[200]={0},phy_add,fno[100];
    struct MainMem mm[100];
    srand((unsigned)time(NULL));
    printf("Enter the Size of File : ");
    scanf("%d",&n);
    printf("Enter the Page Size : ");
    scanf("%d",&s);
    totPages=n/s;
    for(i=0;i<totPages;i++)
    {
        r=rand()%totPages;
        if(arr[r] == 1)
        {
            i--;
            continue;
        }
        arr[r]=1;
        mm[i].pn=r;
        mm[i].start_add=i*s;
        fno[r]=i;
    }
    printf("\n*****\n");
    printf("The Structure of Main Memory\n");
    printf("*****\n");
    printf("Frame\tPage\nNumber\tNumber\n-----\t-----\n");
    for(i=0; i < totPages; i++)
```

```
{
    printf("%d\t%d\n",i,mm[i].pn);
}
printf("*****\n");
printf("Enter The Logical Address\nPage Number : ");
scanf("%d",&pno);
printf("Offset : ");
scanf("%d",&offset);
if(pno >= totPages || offset > s)
{
    printf("Invalid Input\n");
    return 0;
}
phy_add = mm[fno[pno]].start_add + offset;
printf("*****\n");
printf("Physical Address : %d\n",phy_add);
printf("*****\n");
return 0;
}
```

Output:

Output 1:

```
anushka--os@LAPTOP-6G5U0QLQ:~$ ./test
Logical Address To Physical Address
Enter the Size of File : 10
Enter the Page Size : 10

*****
The Structure of Main Memory
*****
Frame    Page
Number   Number
-----
0         0
*****
Enter The Logical Address
Page Number : 0
Offset : 3
*****
Physical Address : 3
*****
anushka--os@LAPTOP-6G5U0QLQ:~$
```

Output 2:

```
anushka--os@LAPTOP-6G5U0QLQ: ~
anushka--os@LAPTOP-6G5U0QLQ:~$ gcc 19BCE0577.c -o test
anushka--os@LAPTOP-6G5U0QLQ:~$ ./test
Logical Address To Physical Address
Enter the Size of File : 100
Enter the Page Size : 10

*****
The Structure of Main Memory
*****
Frame    Page
Number   Number
-----
0         6
1         5
2         3
3         0
4         4
5         7
6         9
7         2
8         1
9         8
*****
Enter The Logical Address
Page Number : 6
Offset : 4
*****
Physical Address : 4
*****
anushka--os@LAPTOP-6G5U0QLQ:~$
```

Output 3:

```
anushka--os@LAPTOP-6G5U0QLQ: ~
anushka--os@LAPTOP-6G5U0QLQ:~$ ./test
Logical Address To Physical Address
Enter the Size of File : 400
Enter the Page Size : 20

*****
The Structure of Main Memory
*****
Frame    Page
Number   Number
-----
0         9
1        19
2        14
3        11
4         3
5        13
6         4
7         1
8        16
9         6
10        10
11         2
12         7
13        12
14        17
15        18
16         0
17         8
18        15
19         5
*****
Enter The Logical Address
Page Number : 16
Offset : 10
*****
Physical Address : 170
*****
anushka--os@LAPTOP-6G5U0QLQ:~$
```