**EXERCISE – 6**

**Name – Bhavy Kharbanda**

**SAP ID – 500082531**

**Roll Number – R214220342**

**Batch – B11**

**Date of Performance – 03/05/2021**

**Date of Submission – 03/05/2021**

OPERATING SYSTEM LAB

EXERCISE – 6.1

MEMORY MANAGEMENT TECHNIQUES

PAGING TECHNIQUE

**AIM: Write a program to implement memory management using Paging technique.**

**Algorithm:-**

**Step1 : Start the program.**

**Step2 : Read the base address, page size, number of pages and memory unit.**

**Step3 : If the memory limit is less than the base address display the memory limit is less than limit.**

**Step4 : Create the page table with the number of pages and page address.**

**Step5 : Read the page number and displacement value.**

**Step6 : If the page number and displacement value is valid, add the displacement value with the address corresponding to the page number and display the result.**

**Step7 : Display the page is not found or displacement should be less than page size.**

**Step8 : Stop the program.**

**Code:-**

#include<stdio.h>

#define MAX 100

int main()

{

int page[MAX],b,i,n,f,p,disp,pno,result;

int choice=0;

printf("\nEnter the base address:");

scanf("%d", &b);

printf("\nEnter the no. of pages in memory: ");

scanf("%d",&n);

printf("\nEnter the page size: ");

scanf("%d",&p);

printf("\nEnter the size of memory unit: ");

scanf("%d",&f);

if(f<b)

printf("\n Memory unit is less than limit");

else

{

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

page[i]=-1;

printf("\nComplete the page table\n");

printf("(Enter frame no as -1 if that page is not present in any frame)\n\n");

printf("\nPage no.\tFrame no.\n-------\t-------");

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

{

printf("\n\n%d\t\t",i);

scanf("%d",&page[i]);

}

do

{

printf("\n\nEnter the Page number and Displacement value:");

scanf("%d%d",&pno,&disp);

if(page[pno]==-1)

printf("\n\nThe required page is not available in any of frames");

else

{

result=(page[pno]\*p)+disp;

printf("\n\nPhysical address is: %d",result);

}

printf("\nDo you want to continue, 1->yes, 0->no?:");

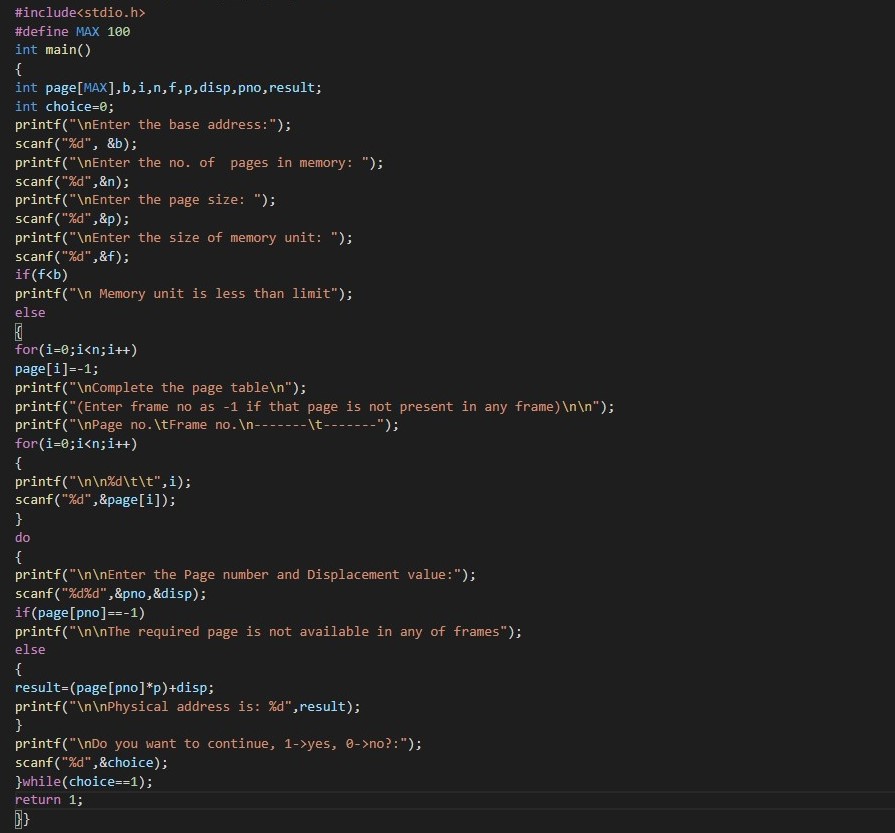
scanf("%d",&choice);

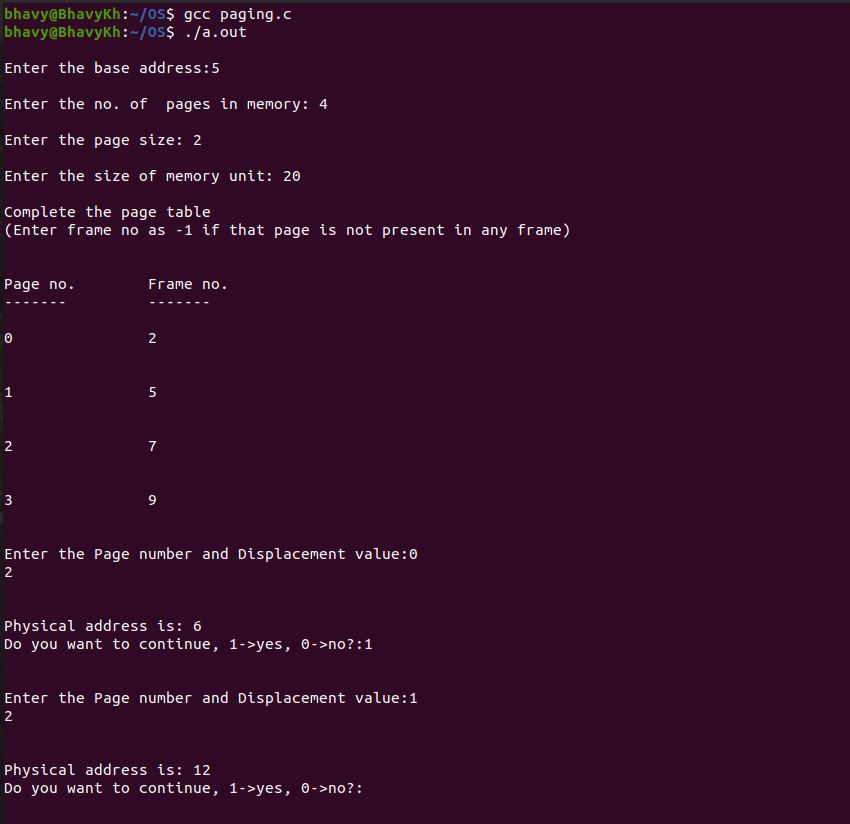
}while(choice==1);

return 1;

} }

**Screenshot/Output:-**





**\*\*\*\*\*\*\*End of Exercise – 6.1\*\*\*\*\*\***

OPERATING SYSTEM LAB

EXERCISE – 6.2

MEMORY MANAGEMENT TECHNIQUES

SEGMENTATION TECHNIQUE

**AIM: Write a program to implement memory management using Segmentation technique.**

**Algorithm:-**

**Step1 : Start the program.**

**Step2 : Read the base address, number of segments, size of each segment, memory limit.**

**Step3 : If memory address is less than the base address display “invalid memory limit”.**

**Step4 : Create the segment table with the segment number and segment address and display it.**

**Step5 : Read the segment number and displacement.**

**Step6 : If the segment number and displacement is valid compute the real address and display the same.**

**Step7 : Stop the program.**

**Code:-**

**#include<stdio.h>**

**#define MAX 100**

**int main()**

**{**

**int segment[MAX],b,i,n,f,p,disp,pno,result;**

**int choice=0;**

**printf("\nEnter the base address:");**

**scanf("%d", &b);**

**printf("\nEnter the no. of Segments in memory: ");**

**scanf("%d",&n);**

**printf("\nEnter the Segment size: ");**

**scanf("%d",&p);**

**printf("\nEnter the size of memory unit: ");**

**scanf("%d",&f);**

**if(f<b)**

**printf("\n Invalid Memory Limit\*\*\*\*\*\*\*\*\*");**

**else**

**{**

**for(i=0;i<n;i++)**

**segment[i]=-1;**

**printf("\nComplete the Segment table\n");**

**printf("(Enter frame no as -1 if that Segment is not present in any frame)\n\n");**

**printf("\nSegment no.\tFrame no.\n-------\t-------");**

**for(i=0;i<n;i++)**

**{**

**printf("\n\n%d\t\t",i);**

**scanf("%d",&segment[i]);**

**}**

**do**

**{**

**printf("\n\nEnter the Segment number and Displacement value:");**

**scanf("%d%d",&pno,&disp);**

**if(segment[pno]==-1)**

**printf("\n\nThe required Segment is not available in any of frames");**

**else**

**{**

**result=(segment[pno]\*p)+disp;**

**printf("\n\nPhysical address is: %d",result);**

**}**

**printf("\nDo you want to continue, 1->yes, 0->no?:");**

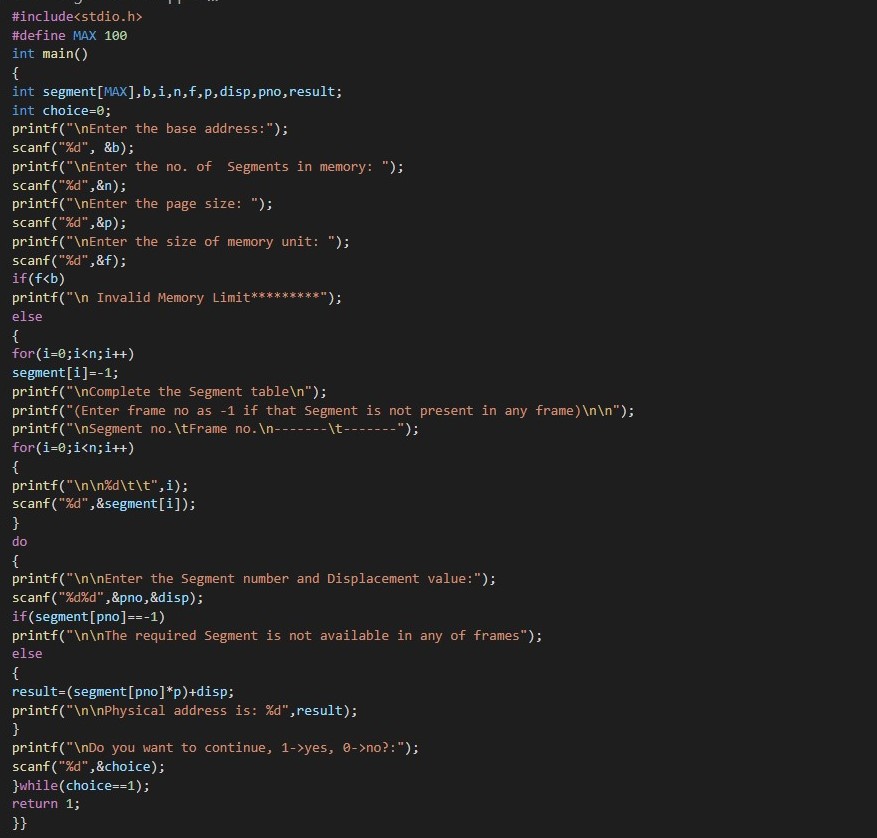
**scanf("%d",&choice);**

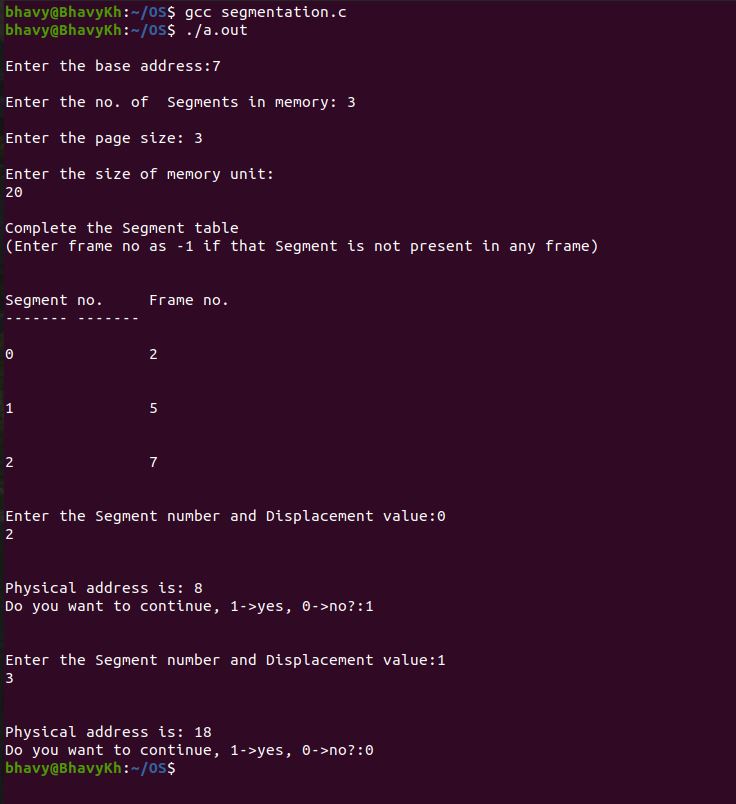
**}while(choice==1);**

**return 1;**

**}}**

**Screenshot/Output:-**





**\*\*\*\*\*\*\*End of Exercise – 6.2\*\*\*\*\*\***

Thank You