S.No: 37	Exp. Name: ***Indexing of a file***	Date:2023-06-24

Aim:

Write a C program to illustrate **Indexing of a file**.
Take an array of integers and find whether the given integer is present or not using **file indexing** method and print the output as shown in the sample output.

Source Code:

fileIndexing.c

```
#include <stdio.h>
#define MAX 25
struct indexfile
{
int indexId;
int kIndex;
};
int main()
{
int numbers[MAX];
struct indexfile index[MAX];
int i,num,low,high,br = 4;
int noOfStudents;
printf("How many numbers do you want to enter:");
scanf("%d",&noOfStudents);
printf("Enter %d numbers:",noOfStudents);
for(i = 0;i < noOfStudents;i++)
{
scanf("%d",&numbers[i]);
}
for(i = 0;i < (noOfStudents/5);i++)
{
index[i].indexId = numbers[br];
index[i].kIndex = br;
br = br + 5;
}
printf("Enter a number to search:");
scanf("%d",&num);
for(i = 0;(i< noOfStudents/5) && (index[i].indexId <= num);i++);
if(i!= 0)
low = index[i-1].kIndex;
else
low = 0;
if(index[i].kIndex != 0&&index[i].kIndex <= noOfStudents)
high = index[i].kIndex;
else
high = noOfStudents;
for(i = low;i <= high;i++)
{
   if(num == numbers[i])
   {
      printf("Number found at position:%d",i);
      return 0;
   }
}
printf("\nNumber not found.");
return 0;
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
How many numbers do you want to enter: 5
Enter 5 numbers: 1 5 6 9 12
Enter a number to search: 6
Number found at position:2
```

Test Case - 2

User Output

```
How many numbers do you want to enter: 7
Enter 7 numbers: 2 3 6 9 12 20 25
Enter a number to search: 20
Number found at position:5
```