Koneru Lakshmaiah Education Foundation

(Deemed to be University)

FRESHMAN ENGINEERING DEPARTMENT

A Project Based Lab Report

On

READING

SUBMITTED BY:

I.D NUMBER	NAME
2200031341	Suvarshitha Turlapati
2200031341	Sruthi Anumolu
2200031389	Chuda Mani Kakarla
2200031606	Pavan Rohit Peteti
2200031608	Balaji Chennupati

UNDER THE GUIDANCE OF

V.PremaLatha

Assistant Professor,CSE.

KL UNIVERSITY

Green fields, Vaddeswaram – 522 502

Guntur Dt., AP, India.

DEPARTMENT OF BASIC ENGINEERING SCIENCES-1

CERTIFICATE

This is to certify that the project based laboratory report entitled "READING" submitted by Mr./Ms**. SUVARSHITHA, SRUTHI, CHUDA MANI, PAVAN ROHIT, BALAJI** bearing Regd. No. 2200031340, 2200031341, 2200031389, 2200031606, 2200031608 to the **Department of Basic Engineering Sciences-1, KL University** in partial fulfillment of the requirements for the completion of a project based Laboratory in "**Computational Thinking for Structural Design** "course in I B Tech I Semester I, is a bonafide record of the work carried out by them under my supervision during the academic year 2022 – 2023.

PROJECT SUPERVISOR

V.PremaLatha

HEAD OF THE DEPARTMENT

Dr. D.Haritha

ACKNOWLEDGEMENTS

It is great pleasure for me to express my gratitude to our honorable President **Sri. Koneru Satyanarayana**, for giving the opportunity and platform with facilities in accomplishing the project based laboratory report.

I express the sincere gratitude to our Director **Dr. A.Jagadeesh** for his administration towards our academic growth.

I express sincere gratitude to HOD-BES-1 **Dr. D.Haritha** for her leadership and constant motivation provided in successful completion of our academic semester. I record it as my privilege to deeply thank for providing us the efficient faculty and facilities to make our ideas into reality.

I express my sincere thanks to our project supervisor V.PremaLatha for her novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to venture this project successfully.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

Project Associates…

ABSTRACT

In the allotted project there is a boy named Vasya . Vasya has got selected to Olympics and started to travel in a train. There he have to travel n hours in the train. He has n hours to travel in the train . in that n hours he has decided to read a book in the train . but he didn't want to read the book continuously for a long time . so he decided to spend k hours out of n hours to read the book . He may not read the book continuously for k hours . Unfortunately, the lighting in the train is not stable. As he is travelling in the way which is completely filled with the tunnels in its way the lighting is not that stable in his way. The brightness inside the train is measured from the integers 0 to 100. Here 0 means very dark and 100 depicts the higher brightness of light inside the train.

INPUT FORMAT:

Input The first input line contains two integers n and k ($1 \le n \le 1000$, $1 \le k \le n$) — the number of hours on the train and the number of hours to read, correspondingly. The second line contains n space-separated integers ai ($0 \le ai \le 100$), ai is the light level at the i-th hour

OUTPUT FORMAT:

In the first output line print the minimum light level Vasya will read at. In the second line print k distinct space-separated integers b1, b2, ..., bk, — the indexes of hours Vasya will read at ($1 \le bi \le n$). The hours are indexed starting from 1. If there are multiple optimal solutions, print any of them. Print the numbers bi in an arbitrary order.

SAMPLE INPUT 1:

5 3

20 10 30 40 10

SAMPLE OUTPUT 1:

20

1 3 4

SAMPLE INPUT 2:

6 5

90 20 35 40 60 100

SAMPLE OUTPUT 2:

35

1 3 4 5 6

INDEX

INTRODUCTION

This project contains an implementation of 'READING' using Dev C++. In this project the user reads the array elements of journey and lighting from the console and find the brightness of the lighting and index values of array. Each integer will be between 1 to 1000, inclusively.

CONCEPTS:

1. FOR LOOP IN C:

Syntax:

```
for (initialization; condition; increment) {
statement(s);
}
```

In this for loop there are three parts- initialization, condition, increment or decrement. Initialization part executes only once.

2. ARRAY IN C:

Array in C can be defined as a method of clubbing multiple entities of similar type into a larger group. These entities or elements can be of int, float, char, or double data type or can be of user-defined data types too like structures. The elements are stored from left to right with the left-most index being the 0th index and the rightmost index being the (n-1) index.

Syntax of array:

```
data_type arrayname[array size];
```

ex: int a[5];

3. CONDITIONAL STATEMENTS-SIMPLE IF:

'if' keyword is used to execute a set of statements when the logical condition is true.

Syntax of simple if:

If condition{

Statements; }

4. .CONDITIONAL STATEMENTS-IF ELSE:

Syntax:

```
if (condition) {
  // block of code to be executed if the condition is true
} else {
  // block of code to be executed if the condition is false
}
```

If the condition is true if part executes and if the condition is false, else part executes. In case of if in place of condition always zero and nonzero value is checked in which means condition is false and nonzero means condition is true.

AIM

To help Vasya choose his reading hours in his allotted k hours of reading in the train out of n hours of his journey in the train. So that this mentioned hours will be comfortable for him to read a book in the train with the comfortable brightness.

Advantages:-

In an array, accessing an element is very easy by using the index number. The search process can be applied to an array easily. For any reason a user wishes to store multiple values of similar type then the Array can be used and utilized efficiently.

Disadvantages:-

- The array is static, which means its size is always fixed. The array is homogeneous, i.e., only one type of value can be store in the array. For example, if an array type "**int**", can only store integer elements and cannot allow the elements of other types such as double, float ,char and so on.
- The array stores data in contiguous(one by one) memory location..

Future Enhancements:-

- To overcome the problem of fixed size of array usage of Dynamic Memory Location like calloc(),malloc()_which helps to reduce wastage of memory by releasing it.
- To overcome the problem of homogenous array, the idea is to structure, where it can store non-homogeneous (heterogeneous) value.
- To overcome the sequential access to the array, the idea is to use the Linked Lists.

SYSTEM REQUIREMENTS

- **SOFTWARE REQUIREMENTS:**

The major software requirements of the project are as follows:

Language : C language

Operating System**:** Windows Xp or later.

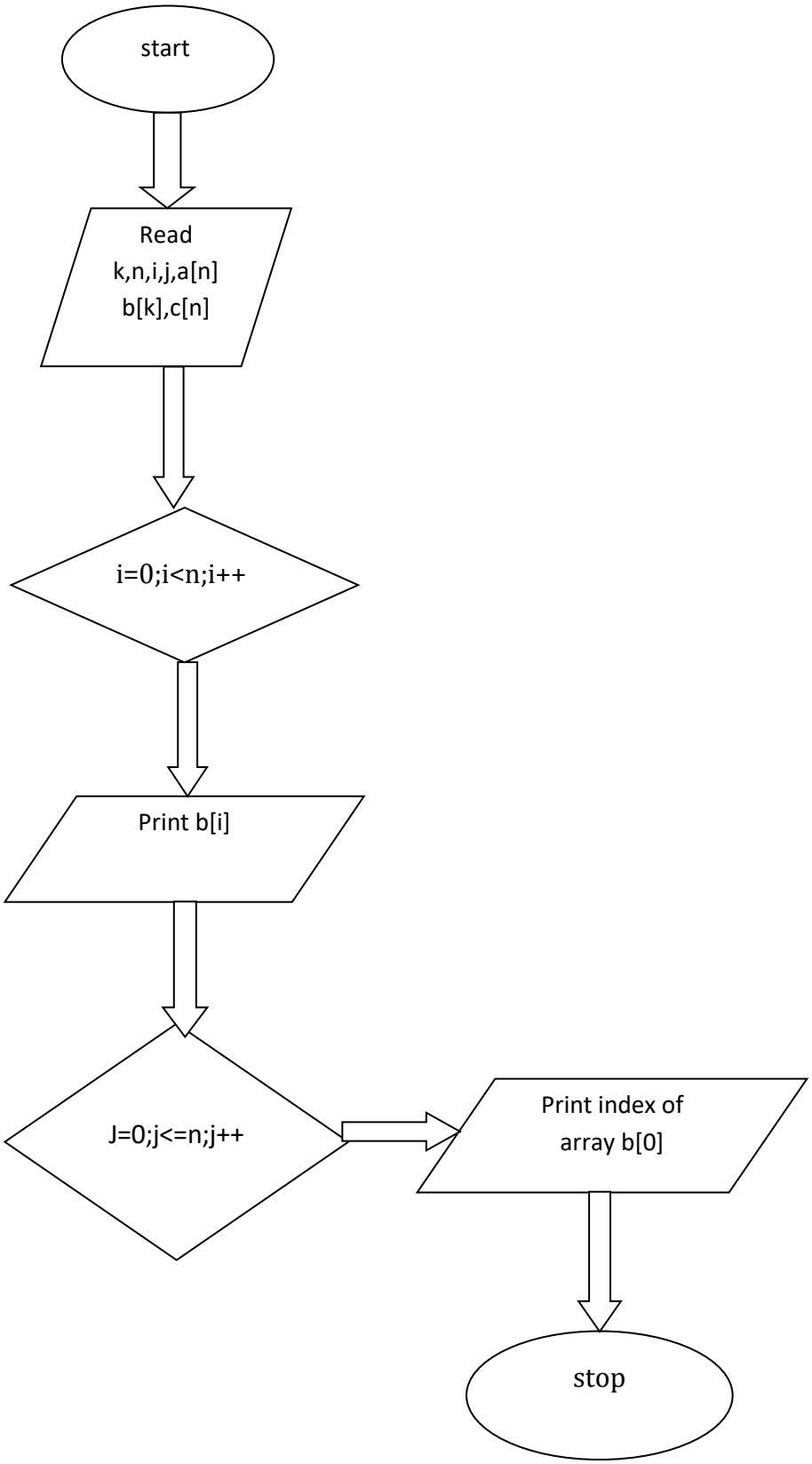
- **HARDWARE REQUIREMENTS:**

The hardware requirements that map towards the software are as follows:

RAM : 16 GB

Processor : : intel core

CLASS DIAGRAM



ALGORITHM

STEP 1 : start

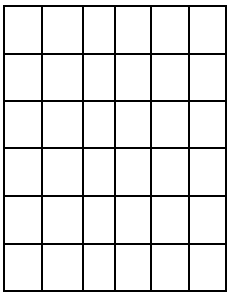
STEP 2 : Read i,j,n[1000], k[1000]

STEP 3 : for(i=0;i<=1;i++)

{

Print no .of hours travelled and no. of hours read

}



no of hours travelled

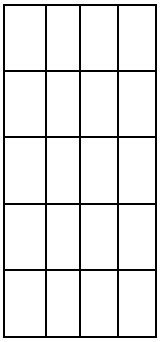
STEP 4 : for(j=o;j<=n;j++)

{

Print index's of array

}

STEP 5 : Stop



No of hours vasya read

IMPLEMENTATION

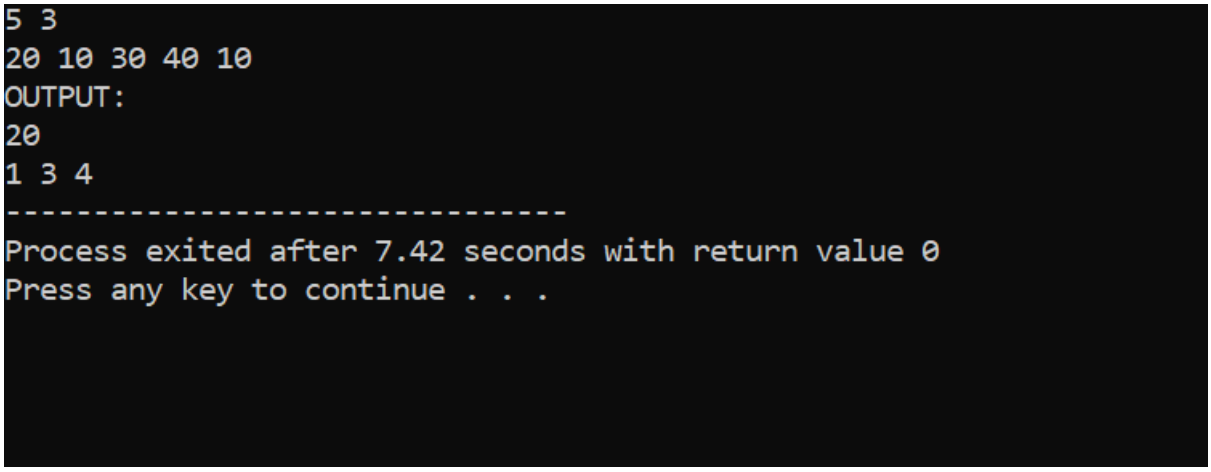
```
#include <stdio.h>
int main()
{
   int n,k,i,j;
   scanf("%d %d",&n,&k);      //n=no.of hrs travelled in train
   int a[n],b[k],c[n];               //no.of hrs vasya read
   for(i=0;i<n;i++)
   {
      scanf("%d", &a[i]);
   }
 for(i=0;i<n;i++)
   {
     c[i]=a[i];
   }
int l=0;
   for (j=0;j<n;j++)
   {
      if(c[j]>c[j+1])
      {
         b[l]=c[j];
         l++;
      }
      else
      {
```

```
        b[l]=c[j+1];
        l++;
      }
  }
 printf("OUTPUT:\n");
  int small=b[0];
  for (i=1;i<k;i++)
  {
     if (b[i]<small)
     {
        small=b[i];
     } }
  printf("%d",small);
  printf("\n");
  for (i=0;i<k;i++)
  {
     for(j=0;j<n;j++)
     {
        if (b[i]==c[j])
        {
           printf("%d ",j+1);
           break;
        }  }
  }
 return 0;
}
```

INTEGRATION AND SYSTEM TESTING

OUTPUTS

Screen Shots:



CONCLUSION

We have attempted to design and implement "Reading". The presence of many built-in class methods takes care of much functionality and reduces the job of coding as well as makes the implementation simpler. We have implemented the project making it as user-friendly and error free as possible. We regret any errors that may have inadvertently crept in.