General Instructions

- This lab test carries 20 marks. The test consists of two questions numbered 1 and 2.
- Programs should be written in C language.
- Assume that all inputs are valid.
- Sample inputs are just indicative.
- Use of global variables is NOT permitted.
- The input should be read from, and the output should be printed to, the console.
- **No clarifications regarding questions will be entertained. If there is any missing data you may make appropriate assumptions and write the assumptions clearly in the design sheet.**
- Solve Part 2 only after submitting a solution (design and code) for Part 1.
- **The students must start with the design of Part 1 and upload the design of Part 1 in the EduServer before 3:30 pm.**
- After uploading the design, students can begin the implementation of Part 1. The source file of Part 1 should be uploaded in the EduServer before 5:00 pm.
- There will be a viva voce during the test.
- Design Submission:
  1. Read the question, understand the problem and write the design (in a sheet of paper) for the indicated function(s) as algorithm(s)(in pseudocode), as per the given prototype.
  2. Take a clear photograph of the handwritten design sheet and submit through the link in eduserver.
  3. The design must be written using pseudocode conventions. There will be a reduction in marks if the student writes C code instead of pseudocode.
  4. For Part 2 also, upload the design first and then start the implementation. Students can upload Part 2 files (design/implementation) till 5:30 pm.
- **The implementation *must be* completely based on the design already submitted.**
- The source code file should be named in the format

  `TEST<NUMBER>_<ROLLNO>_<FIRST-NAME>_<PROGRAM-NUMBER>.c`

  (For example, TEST1_B190001CS_LAXMAN_1.c)

  The source file must be zipped and uploaded. The name of the zip file must be

  `TEST<NUMBER>_<ROLLNO>_<FIRST-NAME>.zip`

  (For example: TEST1_ B190001CS_LAXMAN.zip)

Mark distribution

Maximum marks – 20

- Question 1: 14 Marks (Design - 7 marks, Implementation and Test cases - 5 marks, Viva voce - 2 marks)
- Question 2: 6 Marks (Design - 3 marks, Implementation and Test cases - 3 marks)

1. In a book store, each book is identified using a unique International Standard Book Number ($ISBN$). Given the $ISBN$ and $price$ of n books, store them in two arrays A and B, respectively, each of size n. You may assume that the prices of the books are distinct.

   Write a C program that implements the following functions as per the function prototypes given below.

   - $read(A, B, n)$ - reads $ISBN$ and $price$ of n books store it in array A and B, respectively.
   - $Locate\_ISBN(A, B, n, i)$ – Place a book with $ISBN$ i in the arrays A and B such that:
     - The books that are kept to the left of i are cheaper than i.
     - The books to the right of i are costlier than i the price of i.
     - The relative order of books to the left of i and to the right of i should be same as in the input.
     - Print the location of the book i in the array A, and the details of all the books ($ISBN$ and $price$) after positioning i.

   Input Format

   - First line contains an integer n which is the number of books.
   - Next n lines contain, two integers corresponding to the $ISBN$ and $price$ of a book, separated by a space.
   - The next line contain an integer i, where i is the $ISBN$ of the book to be located in the array A as specified in the function $Locate\_ISBN(A, B, n, i)$

   Output Format

   - First line contains an integer representing the location of the book i.
   - Next n lines containing two integers corresponding to the $ISBN$ and $price$ of a book after the arrangement, separated by a space.

   Sample Input and Output

   Input

```
5
11101 200
11102 1050
11103 150
11104 1250
11105 1010
11102
```

   Output

```
4
11101 200
11103 150
11105 1010
11102 1050
11104 1250
```

2. Use $Locate\_ISBN(A, B, n, i)$ in Qn. 1 to define a recursive function $Find\_Median(A, B, n)$ that prints the $ISBN$ of the book having the median price.
(Median is the $\lceil n/2 \rceil^{th}$ element in a sorted list of numbers.)

Input Format

- First line contains an integer n which is the size of the arrays A and B.
- Next n lines contain, two integers corresponding to the ISBN and price of a book, separated by a space.

Output Format

- A line containing the ISBN of the book having the median price.

Sample Input and Output

Input

```
5
11101 2000
11102 150
11103 1550
11104 1250
11105 100
```

Output

```
11104
```