

Assignment IX

Submit all the programs separately against each assignment in the Moodle System. Provide the result in a separate output file (named, result_<assgn><no>.txt). Use standard output redirection feature to generate the output file.

Hints. If you run the program with the following command

```
./a.out >result.txt
```

Output of your program (generated by printf(.) function) will be written in the file result.txt. You need to provide input from your keyboard, by remembering the sequence of inputs to be given or writing them in a text file in the same sequence.

Otherwise you may use the redirection for the standard input file, such as,

```
./a.out <input.txt
```

For the above all your printing by printf(.) function would be displayed on your monitor.

For both reading from a file and writing to a file use the following.

```
./a.out <input.txt >result.txt
```

If you execute the program multiple times, you may concatenate the outputs in a single file by using the following redirection command:

```
./a.out >>result.txt
```

or

```
./a.out <input.txt >> result.txt
```

(a) Write a function named `getKeyValue(int A[], int n, int key_val, int *pos)` which takes an array of integers `A` of size `n` and a key value `key_val` as argument, it returns 1 if the key value is present in the array, else it returns 0. If the key value is present the smallest array index having the value is returned through another argument variable `pos`.

Write a program which reads `N` integers (to be read from the key-board) in an array. The program reads a sequence of integer values in a while loop and prints the value, if it exists and deletes the element from the array. Otherwise, it prints "The key is not present in the array." The program terminates when the array size becomes 0 with a message, "The array becomes empty", or when the number of failure (not finding a key value) reaches `N` with a message, "The number of failure equals the input size.").

Run your program with the following input data:

(a) `N=10`

Input array of integers = 1, 2, 9, 1, 5, 15, 8, 7, 8, 10.

Sequence of key values to be entered till it terminates:

1, 1, 2, 8, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, (all consecutive numbers)

(b) N=5

Input array of integers: 100, 10, 25, 50, 60

Sequence of key values to be entered till it terminates:

5,10,15,20 (follow the A.P series)

(b) Write a program, by implementing the following functions:

(i) **read_integer_sequence(.)**: It takes a pointer to an integer and a number N as an argument, and dynamically allocate an array of N integers. Then read N integers.

(ii) **swap(.)**: It exchanges values of two integer variables whose addresses are passed as arguments in the function.

(iii) **sort_integer_sequence(.)**: It takes an array of integers and its number of elements as arguments, and returns the array sorted in ascending order by using the function **swap(.)**.

The program reads two separate integer sequences of sizes L (to be read) and M (to be read), respectively using the above function, and prints the number and the list of distinct elements in common to both the sequence in ascending order. For sorting in ascending order the program uses the function **sort_integer_sequence(.)**. If no such element exists, it prints a message 'No distinct common element found'.

For example, consider the following two sequences of integers as input.

L=8

55 20 12 36 76 -20 0 12

M=6

12 82 28 35 12 -20

The program prints:

Number of common distinct elements: 2

They are: -20 12

Provide output for the following input data.

(i) L=10

7 -8 16 89 24 86 -8 12 33 56

M=9

3 48 -8 86 56 -8 12 0 1

(ii) L=10

7 -8 16 89 24 86 -8 12 33 56

M=9

13 148 -18 186 156 -18 112 10 11