Informatik für Mathematiker und Physiker - AS18

Exercise 7: References and Vectors

Handout: 30. Okt. 2018 06:00

Due: 5. Nov. 2018 23:59

Task 3: Longest Increasing Subsequence

Open Task (https://expert.ethz.ch/solve/5hZcQQrdzKBs79T7a)

Task

Write a program using a vector to perform the following:

1. Read integer values until a negative one is found
2. Output the length of the longest increasing subsequence within the input sequence (excluding the terminating negative integer).

In other words, your program should find the maximum number of consecutively increasing value in the array. We only consider strict increase here, for example, "4 4" is not an increasing sequence.

Your solution should split the sub-tasks of the program (scanning the input and computing the result, and maybe other sub-tasks) into separate functions. Those functions should use reference parameters with the right constness to access the vector.

Note: We consider solutions as incorrect if they use library functions (or data structures) beyond the ones discussed in the lectures so far to perform the actual computation.

Input

A sequence of of integer values terminated by a single negative integer

Example:

```
22 35 4 16 42 3 -1
```

Output

The length of the longest increasing subsequence of the input sequence, not counting the terminating negative integer.

Example: for the input given above, the longest increasing subsequence is `4 16 42`, so the result should be

```
3
```

Task 2: Number of Occurrences

Open Task (https://expert.ethz.ch/solve/DzxdaN3NabrggGJfe)

Task

Write a program using a vector to perform the following:

1. Read integer values until a negative integer is encountered;
2. Read an extra non-negative integer `n` ;
3. Output the number of occurrences of `n` encountered during phase (1).

Note: We consider solutions as incorrect if they use library functions (or data structures) beyond the ones discussed in the lectures so far to perform the actual computation.

Input

A sequence of of integer values containing a single negative integer, in penultimate position.

Example:

```
0 1 1 2 3 5 8 13 21 -1 1
```

Output

The number of occurrences of the last integer in the sequence before the negative integer.

Example: for the input given above, the number of occurrences of 1 before -1 occurs is

```
2
```

Task 1: Const and reference types

Open Task (https://expert.ethz.ch/solve/GzXNTmAMnNmczDDh8)

This task is a text based task. You do not need to write any program/C++ file: the answer should be written in main.txt *(and might include code fragments if questions ask for them).*

Task

Consider the following family of functions:

```
T foo (S i)
{
    return ++i;
}
```

with `T` being one of the types `int`, `int&`, and `const int&`, and `S` being one of the types `int`, `int&`, `const int`, and `const int&`. This defines 12 different functions, and all of those combinations are syntactically correct.

For the following combinations:

1. `T=int`, `S=int`

2. `T=int`, `S=const int`

3. `T=int`, `S=int&`

4. `T=int&`, `S=int`

5. `T=const int&`, `S=int&`

answer the following questions:

- a) Is the function definition is *semantically valid*, i.e., is the constness of variables and references respected? If not, give the reason why (max. 15 words).

- b) If the function is semantically valid: Is the resulting function definition also *valid during runtime*, meaning that function calls always have well-defined value and effect? If not, give the reason why (max. 15 words).

- c) If the function is semantically valid and valid during runtime: Give the precise post condition for the corresponding function `foo`.

Task 4: Sorting by Swapping

Open Task (https://expert.ethz.ch/solve/uQYY73CrBjxpxfjR7)

Write a program that performs the following:

1. Read a non-negative integer n
2. Read n integers
3. Sort the n integers read in (2) in decreasing order. To that end, we suggest the following method:
    1. Store the n integers in a vector
    2. Iterate through the vector, compare each pair of adjacent values and swap them if they are in the wrong order (i.e increasing order)
    3. Repeat step (2) as long as swap are performed.
4. Output the sorted values.

Your solution should isolate the sorting functionality of the program into a separate function, using a reference parameter with the right constness to access the vector. Other sub-tasks of the program may be isolated as well.

Note: We consider solutions as incorrect if they use library functions (or data structures) beyond the ones discussed in the lectures so far to perform the actual computation. In particular, library sorting functions are forbidden as they defeat the purpose of the exercise.

Input

A sequence of integers, preceded by its length.

Example:

```
7 3 8 6 3 4 6 5
```

Output

The input sequence, minus the length, in decreasing order. The integers are separated by a single space.

```
8 6 6 5 4 3 3
```