Exercise 0

Damjan Kostovic

25 February 2022

About Me

Exercise Classes (Lab)

- Weekly on Fridays 10:15 - 11:45, starting from February 25 2022.
- 13 Labs in total
- Place: Zoom

All of my labs are on on Zoom. Recordings will be uploaded to OLAT.

Forums on OLAT

▶ Go to OLAT Forums to ask questions.
"Exercise" tab is for Exercises.
"Lab" tab is for your labs.
"General Forum" is for general questions.

My Suggestions for the Course

- ▶ Attend the labs.
- ▶ Practice as much as you can.
- ▶ Solve (or try to solve) the exercises multiple times!!
- ▶ Don't neglect the course until 2 weeks before the exam (really!).

What the goal of this lecture is (in my opinion)

- ▶ Algorithmic thinking (needed everywhere).
- ▶ C.
- ▶ Being able to solve difficult problems on your own.
- ▶ If you pass this course, you'll be fine in all other courses.
- ▶ Bonus: these kinds of questions are often asked at interviews at large tech companies.

Some C basics

- ▶ Every statement in C has to end with a semicolon (;)
- ▶ Operators & &, || and ! (instead of "and", "or" and "not" as in Python)
- ▶ Blocks are indicated with curly braces, additional whitespaces are ignored (but should be properly set either way).
- ▶ Comments are indicated with /*, */ (and //) for comments (instead of # as in Python).
- ▶ Functions have return types (e.g. int) but there is no def keyword.
- ▶ The function printf is used for output to the console (and not print as in Python).
- ▶ Libraries are made available using include (instead of import as in Python).
- ▶ ...

More C basics

- C uses static type checking, meaning that you as programmers have to specify before runtime what data type a variable will have and this data type will then stick unchangeably to that variable. Python in contrast uses dynamic type checking.
- C is not an object-oriented programming language

Find the second largest integer in an array of integers.

Input: An array $A[1..n]$ with n integers, where $n \geq 2$.
Output: the second largest integer in A.
Solutions are not unique.

Task 2 solution in pseudocode

1 **Algorithm:** SecondLargest(A[1..n])

2 pos1 = -1; pos2 = -1;

3 target = MIN;

4 **for** $i = 1$ **to** n **do**

5   **if** $A[i] > target$ **then**

6 | target = A[i]; pos1 = i;

7 target = MIN;

8 **for** $i = 1$ **to** n **do**

9   **if** $i \neq pos1 \wedge A[i] > target$ **then**

10 | target = A[i]; pos2 = i;

11 **return** A[pos2];

Task 2 solution in C code

```c
#include <stdio.h>
#define MIN -999999999

int main() {
  int n = 5;
  int A[] = { 11, 3, -3, 2, -5};
  int pos1 = -1, pos2 = -1;
  int target = MIN;
  for(int i = 0; i < n; i++ ) {
    if(A[i] > target) {
      pos1 = i;
      target = A[i];
    }
  }
  target = MIN;
  for(int i = 0; i < n; i++ ) {
    if(i != pos1 && A[i] > target) {
      pos2 = i;
      target = A[i];
    }
  }
  printf("%d\n", A[pos2]);
}
// gcc ex0.c -o ex0; ./ex0
```