Informatics II, Spring 2024, Exercise 0

Publication of exercise: February 16, 2024
Publication of solution: no solutions
Exercise classes: February 19 - 23, 2024

During the exercise class you will perform the necessary installations and then write and run a simple C program. Doing this is important so you are prepared for the exercises in the upcoming exercise classes.

Task 1: Install gcc compiler

Windows: Follow the *installation* instructions shown in `https://www.msys2.org`.

Linux and MacOS: The gcc compiler should already be installed.

Run the command "gcc -v" in the terminal. If you have installed gcc correctly, it shows the version of gcc.

Task 2: Compile and run the search.c file

- Go to the folder that contains the "search.c" file in the terminal.
- Run "gcc search.c -o search" in terminal.
- Run "./search" in terminal.
- Explain what the program outputs.

Task 3 [Optional]

The goal of this task is find the second largest integer in an array of integers.

Input: An array $A[1..n]$ with n distinct integers, where $n \geq 2$.
Output: the second largest integer in A.

Write a pseudocode algorithm as well as a C code program for this task. Compare the differences between the C code program and its pseudocode counterpart.

Solution: The sample solution shows a simple two-pass solution for this task. More often than not, you first find a simple solution and then improve it.

Two-pass C code solution:

```
#include <stdio.h>
```

```
int main() {
  int n = 5;
  int A[] = { 11, 3, -3, 2, -5};

  int pos1 = 0;
  int temp1 = A[pos1];
  for(int i = 0; i < n; i++ ) {
    if(A[i] > temp1) {
      pos1 = i;
      temp1 = A[i];
    }
  }

  int pos2;
  if(pos1 == 0) {
    pos2 = 1;
  } else{
    pos2 = 0;
  }
  int temp2 = A[pos2];
  for(int i = 0; i < n; i++ ) {
    if(i != pos1 && A[i] > temp2) {
      pos2 = i;
      temp2 = A[i];
    }
  }
  printf("%d\n", A[pos2]);
}
```

Two-pass pseudocode solution:

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

```
pos1 = 1;
temp1 = A[pos1];
for i = 1 to n do
    if A[i] > temp1 then
        temp1 = A[i];
        pos1 = i;
if pos1 == 1 then
    pos2 = 2;
else
    pos2 = 1;
temp2 = A[pos2];
for i = 1 to n do
    if i ≠ pos1 ∧ A[i] > temp2 then
        pos2 = i;
        temp2 = A[i];
return A[pos2];
```