

# Algorithms & Data Structures: Lab 02

week of 8th October 2018

This week's lab includes an assessment, available from the module learn.gold page, which is open only during this week. Aim to submit before the end of the session!

## 1 Setup

### 1.1 Saving your work

You might by now have used `git` to download the lab bundle, to test your setup on your own computer. (If you haven't, but are planning to work on lab computers, you will nevertheless need to understand this section in order to save your work in future.) You might have made changes to your copy, and at this point you should save those changes. First, examine your copy to see if there are any changes; run these commands from the labs directory:

```
git status
git diff
```

The first command will summarize files that have changed in the directory relative to the pristine copies; the second will show you the details of the changes. If you are satisfied that you want to keep the changes, store them in your local version control system by doing

```
git commit -a
```

and writing a suitable commit message. (If you have no changes, you don't need to do this)

### 1.2 Downloading this week's distribution

Once you have successfully saved your changes from last week, if any, you can get my updates by doing

```
git pull
```

which *should* automatically merge in new content. After the `git pull` command, you should have a new directory containing this week's material (named 02/) alongside the existing 00/ and 01/ directories.

Alternatively, you can make a fresh download of everything, for example to a different computer, using

```
git clone http://gitlab.doc.gold.ac.uk/crhodes/is52038b-labs.git
```

which will check out the lab bundle to a directory called `is52038b-labs` under your current working directory.

## 2 Pseudocode exercises

### 2.1 Exercise 1

Consider the following pseudocode, defining a function EXERCISE1:

```
1: function EXERCISE1(v)
2:   a  $\leftarrow$  0; b  $\leftarrow$  0
3:   for  $0 \leq i < \text{LENGTH}(v)$  do
4:     if  $v[i] > b$  then
5:       if  $v[i] > a$  then
6:         b  $\leftarrow$  a
7:         a  $\leftarrow$  v[i]
8:       else
9:         b  $\leftarrow$  v[i]
10:      end if
11:    end if
12:  end for
13:  return b
14: end function
```

Run EXERCISE1 (recording the states of the program on paper or in a text editor) on the following inputs:

- the vector {5,2,0,3,8}
- the vector made up of your birth date (e.g. {1,9,7,8,1,2,2,5})

Exchange your answers with your neighbour, and discuss:

1. what do you think this pseudocode does?
2. can you prove it?
3. in terms of the length of v, how many times does line 4 get executed? What about line 6?

## 2.2 Exercise 2

Consider the following pseudocode, defining a function EXERCISE2:

```
1: function EXERCISE2(v)
2:   for 0 < i < LENGTH(v) do
3:     current ← v[i]
4:     j ← i - 1
5:     while j ≥ 0 do
6:       if v[j] ≤ current then
7:         break
8:       end if
9:       v[j+1] ← v[j]
10:      j ← j - 1
11:    end while
12:    v[j+1] ← current
13:  end for
14:  return v
15: end function
```

Run EXERCISE2 (recording the states of the program on paper or in a text editor) on the following inputs:

- the vector {5,2,0,3,8}
- the vector made up of the digits of your birth year (e.g. {1,9,7,8})

Exchange your answers with your neighbour, and discuss:

1. what do you think this pseudocode does?
2. can you prove it?
3. in terms of the length of v, how many times does line 4 get executed? What about line 6?

## 3 Hello, world

Your lab bundle should contain a directory named 02, with subdirectories cpp and java for C++ and Java respectively. Your task for this part of the lab is to implement two methods:

1. `studentNumber()`, which should return your student number (as an integer)
2. `moodleID()`, which should return your Moodle ID number, which you can find by clicking on your name in the footer of every learn.gold page while you are logged in and looking at (and expanding if necessary) the URL bar of your browser.

Check that you can compile your modified code by running `make` in the appropriate directory. The test cases provided with the lab bundle (that you can run using `make test`) do not check that you have correctly identified these numbers for yourself: only that they are plausible.

## 4 Uploading your work

Before the end of the session, you must submit your work to the online submission system. Access to the online submission will be closed at **16:00 on Friday 12th October**. You may submit more than once, and your best score will be kept: do not wait until the very end of the week to submit.