

Test Pearl 001 — Algorithmics

Pearls of Computer Science (201300070) / Introduction to BIT (201300073)

16 september 2016, 13:45–14:45

Module coordinator: Maurice van Keulen

Instructor: Arend Rensink

- You may use 1 A4 sheet with your own notes for this test, as well as a *simple* calculator
 - Scientific or graphical calculators, laptops, mobile phones, books etc. are not allowed.
- Put those in your bag now (with the sound switched off)!**

- Total number of points: 100

15 points **Question 1** Suppose that you execute the following assignments in Python:

```
student = ["Al", "Becca", "Charlie"]
grade = [8, 9, 6]
```

student is a list of threestudents of Pearl 001, whereas grade is a list of their corresponding test grades.

- Write a Python condition (*not* an entire `if`-statement) that tests whether student number `i` has the highest grade of the three. (You do not know the value of `i`.)
- Write a Python assignment that assigns to a new variable `p` the pair consisting of elements `i` of `student` and `grade`.
- Write a sequence of assignments that is as short as possible, resulting in a change to `student` after which the student names are ordered from highest to lowest grade. (It is *not* correct to assign an entirely new value to `student`: you must *modify* the list by swapping elements.)

15 points **Question 2** Analyse the following Python-function. (Remember that `//` stands for *integer division*, which ignores the remainder: for instance, `7 // 2 == 3`.)

```
1 def compute(data):
2     i = 0
3     while i < len(data)//2:
4         print("i=", i, ", data=", data)
5         temp = data[len(data)-i-1]
6         data[len(data)-i-1] = data[i]
7         data[i] = temp
8         i = i+1
9     return data
```

- Show what happens in a call of `compute(["Some", 1, "mixed", 3, "list"])`, by writing down the output of the `print`-statement in line 4 every time it is executed. What is the return value of the call?
- What does the function `compute` actually do? (*Do not give a step-by-step explanation of the execution, but describe the purpose of the function.*)

10 points **Question 3** How many steps does `compute` (in Question 2) need in terms of the length (say n) of the input list `data`: approximately (“in the order of”) $\log_2 n$, approximately \sqrt{n} , approximately n or approximately n^2 ? Explain your answer.

10 points **Question 4** What would happen if the comparison `i < len(data)//2` in line 3 of `compute` (Question 2) would be replaced by `i < len(data)`? Explain your answer.

20 points **Question 5** Consider the list `[10, 0, 8, -3, 15, 5]`.

- Show how bubble sort sorts this list, by writing down the list after every single modification.
- Show how merge sort sorts this list, by schematically showing how the list is split and zipped back together.

10 points **Question 6** One night you can't sleep after having prepared yourself all evening for the Algorithmics test, and you wonder if one could not get an even faster *ternary search* algorithm by dividing the list into *three* equal parts and determining in which of those three parts (left, middle or right) the searched-for value lies.

Can this idea be made to work, i.e., could such an algorithm work in practice? If not, why not? If so, what would be the complexity of ternary search — in particular, would it be fundamentally faster than binary search? Explain your answer.

20 points **Question 7** You have a large bowl containing thousands of marbles of an incredible variety, from which you want to pick two identical marbles to give to your twin nieces at their birthday. You don't care which two. You are not even sure that there *are* two identical marbles in the bowl. How do you go about this?

- (a) Write an algorithm in natural language, with unambiguous, numbered instructions, that allows you to reach your goal in as few steps as possible. Your instructions may only involve one or two marbles at a time, never an arbitrary set of them.

You may assume that you have ample space to arrange the marbles in any way you like. You do not have to tidy up afterwards.

Do not try to give an answer in Python!

- (b) How many steps does your algorithm need in the worst case, as a function of the number of marbles in the bowl?