University of Bergen
Faculty of Mathematics and Natural Sciences

Exam in subject INF102 — Algorithms, data structures and programming

Friday November 28, 09:00–12:00

English version

No auxiliary resources are allowed

Task 1 (20%)

In this task we will have a look at runtime.

Assume that n is a positive integer. What is the runtime of the following code snippets? Give the runtime as order of growth. Support your answers with explanatory text.

(a)

```
s = 0;
for (i = 1; i < n; i++)
  for (j = i; j <= n; j++)
    s = s + 1;
```

(b)

```
c = 0;
for (i = 1; i < n; i++) {
  for (j = i; j < n && j%10 != 3; j++) {
    c++;
  }
  i = j-1;
}
```

(c)

```
for (i = 0; i < n; i = i + 2) {
  j = 0;
  while (j < i)
    j = j + 1;
  k = 0;
  while (k < i)
    k = k + 1;
}
```

(d)

```
for (i = 0; i < n; i++) {
  j = i;
  while (j > 0)
    j = j / 2;

}
```

(e)

```
s = 0;
for (i = 1; i <= n; i++)
  for (j = 2; j <= n; j++)
    for (k = 3; k <= n; k++)
      s = i + j + k;
```

Task 2 (20%)

In this task we will have a look at sorting algorithms.

(a) Explain how insertion sort and merge sort work.

(b) What are the worst-case runtimes of these algorithms? Support your answer with explanatory text.

(c) Give a trace of insertion sort on the following string: "magnus". Sorting is to be done in standard lexicographic order.

(d) Do the same for merge sort.

Task 3 (30%)

In this task we will discuss data structures for the representation of a set data type. The elements of our sets are ints, and each element can occur at most once in a set. The API of our data type is as follows:

```
public class Set

        Set()           // create an empty set
   void insert(int x)   // insert element x
boolean contains(int x) // return true if x is in the set,
                        // false otherwise
```

(a) List three different ways in which our set data type could be represented.

(b) Evaluate these data structures with respect to whether they are appropriate for an efficient implementation of the Set API and discuss your findings.

Task 4 (30%)

In this task you are to consult the nation Islandia with your superb algorithm skills. The country consists of a bunch of islands, N of them to be more specific. Back in the old days, the Islandians used rowing boats to get back and forth, selling their goods and visiting their families. However, times have changed, and now people want to travel with their cars, segways and what not. As a consequence, the country is in desperate need of bridges and the goal is to make it possible to travel between every pair of islands without using boats. Some bridges have already been built, while others have been evaluated by the newly created ministry of bridges. Provided with the information of what bridges have been built and what bridges could be built, you are to answer some of the questions of the

ministry. More specifically, the islands are numbers from 0 to $N-1$ (as you advised them to, being a computer scientist and all). For each of the bridges that are already built, you are given the two islands it connects. And for each of the prospective bridges, you are given the two islands it would connect and the price of building that specific bridge.

(a) Describe an algorithm that decides whether it is at all possible to reach the goal of connecting all islands by building some or all of the prospected bridges.

(b) Assuming that the algorithm in (a) concluded that it is indeed possible to reach the goal of connection: describe an algorithm that computes the minimum amount of money Islandia must spend to connect all islands.

(c) We will now consider the case that the algorithm in (a) concluded that the prospective bridges are not sufficient to connect all islands. Describe an algorithm that gives a minimum number of new bridges that should be considered such that it would be possible to connect all islands.

Give a well-formulated description of the algorithms and argue for each algorithm why it is correct. If you use an algorithm from the curriculum, you do not have to argue its correctness, but its appropriateness. In addition, explain the worst case runtime of your algorithms. To obtain full score, the algorithms in (a) and (c) should have $O(N+B+P)$ worst case runtime and the algorithm in (b) should have $O((N+B+P)\log N)$ worst case runtime were N is the number of islands, B the number of already built bridges and P the number of prospective bridges.

Good luck! Markus Sortland Dregi and Marc Rehmsmeier