CSCM12 – Coursework I

Arno Pauly

a.m.pauly@swansea.ac.uk

- By submitting this work, you state that you fully understand and are complying with the university's policy on Academic Integrity and Academic Misconduct. The policy can be found at https://www.swansea.ac.uk/academic-services/ academic-guide/assessment-issues/academic-integrity-academic-misconduct.
- This is an individual assignment, and you must not collaborate with others or share solutions.
- If you are using sources other than the lecture material, cite them.
- You submit your solutions by uploading a single pdf file on Canvas.
- Other than drawings (for Exercise 3), your answers need to be **typed**. Drawings can either be created by hand, or by suitable software tools.
- There are 30 marks to be earned in total. Each of lab sheets 1-4 earns you 1 bonus mark (but 30 total is the maximum achievable).

Exercise 1 (8 marks)**.** For each of the following functions, determine their asymptotic complexity. Justify your answer.

1.
```
static int bla1(int n) {
    int result = 0;
    for (i=0,i++,i<n) {
       result += i;
    }
    return result;
}
```

2.
```
static int bla2(int n) {
    int result = 0;
    for (i=0,i++,i<n) {
       result += bla1(n);
    }
    return result;
}
```

3.
```
static int bla3(int n) {
   if (n <= 0) {
      return 0;
   }
   else {
      return bla1(n) + bla3(n-1);
   }
}
```

4.
```
static bool bla4(int n) {
   if (n == 1) {
      return true;
   }
   elseif (n % 2 == 1) {
      return false;
   }
   else {
      return bla4(n /2);
   }
}
```

Exercise 2 (6 marks total)**.** Consider the following function:

```
static int bla5(int n) {
   if n <= 1 {
      return n
   }
   else {
      return 2*bla5(n-1) - bla5(n-2);
   }
}
```

a) What is the asymptotic complexity of `bla5`? (2 marks)

b) Write a function in Java or pseudocode that computes the same values as `bla5`, but which runs in time $\mathcal{O}(n)$. (4 marks)

Exercise 3 (6 marks)**.** Illustrate how mergesort works on one of the following inputs, depending on the last digit of your student number.

1. If the digit is 0 or 1, use: 13 15 7 6 9 3 1 4 16 12 5 8 2 14 11 10
2. If the digit is 2 or 3, use: 10 1 12 6 7 14 13 8 9 3 16 15 4 5 11 2
3. If the digit is 4 or 5, use: 10 16 8 12 5 11 13 4 3 2 6 9 1 14 7 15
4. If the digit is 6 or 7, use: 13 2 14 16 6 15 4 3 9 10 7 8 5 11 1 12
5. If the digit is 8 or 9, use: 3 14 11 1 9 6 7 5 15 10 13 2 4 12 16 8

Exercise 4 (8 marks total)**.** a) Explain how we can view an instant elimination tournament as a recursive algorithm to identify the best team. (3 marks)

b) Assume you have a function `static bool match(team a, team b)` which returns true if team a wins, and `false` if team b wins. There are no draws. Write a recursive function of type `static team tournanmentwinner(team arg[])` in either Java or pseudocode that uses the tournament idea to determine the winner of the tournament. (3 marks)

c) Contrast your function with the following algorithm to determine the best team. What is the downside of this one for actual tournaments? (2 marks)

```
static team tournanmentwinner2(team arg[]) {
   bestteam = arg[0];
```

```
        for (i=1,i++,i<arg.length()) {
            if match(arg[i],bestteam) {
                bestteam = arg[i];
            }
        }
        return bestteam;
    }
```