Exercise Sheet 2
COMS10017 Algorithms 2020/2021

Reminder: $\log n$ denotes the binary logarithm, i.e., $\log n = \log_2 n$.

1 Θ and Ω

1. Prove that the following two statements are equivalent:
   (a) $f \in \Theta(g)$.
   (b) $f \in O(g)$ and $g \in O(f)$.
2. Prove that the following two statements are equivalent:
   (a) $f \in \Omega(g)$.
   (b) $g \in O(f)$.
3. Let $c > 1$ be a constant. Prove or disprove the following statements:
   (a) $\log_c n \in \Theta(\log n)$.
   (b) $\log(n^c) \in \Theta(\log n)$.
4. Let $c > 2$ be a constant. Prove or disprove the following statement:

$$2^n \in \Theta(c^n) \ .$$

2 O-notation

1. Consider the following functions:

$$f_1 = 2^{\sqrt{n}}, f_2 = \log^2(20n), f_3 = n!, f_4 = \frac{1}{2}n^2/\log(n), f_5 = 4\log^2(n), f_6 = 2^{\sqrt{\log n}} \ .$$

   Relabel the functions such that $f_i \in O(f_{i+1})$ (no need to give any proofs here).
2. Give functions f, g such that $f(n) \in O(g(n))$ and $2^{f(n)} \notin O(2^{g(n)})$.

3 Runtime Analysis

Algorithm 1

Require: Int $n \geq 1$
  $x \leftarrow 0$
  for $i = 1 \ldots n$ **do**
    for $j = 1 \ldots n$ **do**
      $x \leftarrow x + i \cdot j$
    end for
  end for
  return x

Algorithm 2

Require: Int $n \geq 1$
  $x \leftarrow 0$
  for $i = 1 \ldots n$ **do**
    for $j = i \ldots n$ **do**
      $x \leftarrow x + i \cdot j$
    end for
  end for
  return x

Algorithm 3

Require: Int $n \geq 1$
  $x \leftarrow 0$
  $i \leftarrow 1$
  while $i \leq n$ **do**
    for $j = 1 \ldots n$ **do**
      $x \leftarrow x + i \cdot j$
    end for
    $i \leftarrow 2 \cdot i$
  end while
  return x

Algorithm 4

Require: Int $n \geq 1$
  $x \leftarrow 0$
  $i \leftarrow 1$
  while $i \leq n$ **do**
    for $j = 1 \ldots i$ **do**
      $x \leftarrow x + i \cdot j$
    end for
    $i \leftarrow 2 \cdot i$
  end while
  return x

Determine the runtimes of Algorithms 1,2,3 and 4 using Big "Theta" notation.

4 Optional and Difficult Questions

Exercises in this section are intentionally more difficult and are there to challenge yourself.

4.1 Average Case Runtime of Linear Search

For integers $k, n \geq 1$ let $S_k(n)$ be the set of all integer arrays of length n where every array entry is taken from the set $\{0, 1, 2, \ldots, k-1\}$.

1. What is the average case runtime of linear search on $S_3(n)$?
2. What is the average case runtime of linear search on $S_C(n)$, for any constant C?
3. What is the average case runtime of linear search on $S_n(n)$?
4. What is the average case runtime of linear search on $S_{\sqrt{n}}(n)$?