Comp 251 (Fall 2022): Assignment 1

Answers must be submitted online by October 15th (11:59 pm), 2022.

General instructions (Read carefully!)

- **Important:** All of the work you submit must be done by only you, and your work must not be submitted by someone else. Plagiarism is academic fraud and is taken very seriously.
- To some extent, collaborations are allowed. These collaborations should not go as far as sharing code or giving away the answer. **You must indicate on your assignments the names of the people with whom you collaborated or discussed your assignments (including members of the course staff). If you did not collaborate with anyone, write "No collaborators". If asked, you should be able to orally explain your solution to a member of the course staff.**
- It is your responsibility to guarantee that your assignment is submitted on time. We do not cover technical issues or unexpected difficulties you may encounter. Last minute submissions are at your own risk.
- Multiple submissions are allowed before the deadline. We will only grade the last submitted file. Therefore, we encourage you to submit as early as possible a preliminary version of your solution to avoid any last minute issue.
- Late submissions can be submitted for 24 hours after the deadline, and will receive a flat penalty of 20%. We will not accept any submission more than 24 hours after the deadline. The submission site will be closed, and there will be no exceptions, except medical.
- In exceptional circumstances, we can grant a small extension of the deadline (e.g. 24h) for medical reasons only. However, such request must be submitted before the deadline, justified and approved by the instructors.
- Violation of any of the rules above may result in penalties or even absence of grading. If anything is unclear, it is up to you to clarify it by asking either directly the course staff during office hours, by email at (`cs251@cs.mcgill.ca`) or on the discussion board on Ed (recommended). Please, note that we reserve the right to make specific/targeted announcements affecting/extending these rules in class and/or on the website. It is your responsibility to monitor Ed for announcements.
- The course staff will answer questions about the assignment during office hours or in the online forum. We urge you to ask your questions as early as possible. We cannot guarantee that questions asked less than 24h before the submission deadline will be answered in time. In particular, we will not answer individual emails about the assignment that are sent sent the day of the deadline.
- Unless specified, **all answers must be justified!**

1. The algorithm of Euclid computes the greatest common divisor (GCD) of two integer numbers a and b. The following pseudo-code is the original version of this algorithm.

Algorithm 1 Euclid1(a,b)

Require: $a, b > 0$
Ensure: $a = GCD(a, b)$
  while $a \neq b$ **do**
    if $a > b$ **then**
      $a \leftarrow a - b$
    else
      $b \leftarrow b - a$
    end if
  end while
  return a

We want to prove the correctness of this algorithm using the loop invariant technique.

(a) (10 points) In a first attempt to prove the correctness of this algorithm, we propose the following loop invariant property: "$GCD(a, b)$ is a factor of a and b". Explain why this property will not help you as is.

(b) (10 points) Use your previous observation to propose another loop invariant property that will help you to prove the correctness of the version of Euclid's algorithm presented above.

(c) (10 points) Show that your loop invariant property is true before executing the While loop for the first time (i.e., initialization property).

(d) (10 points) Then, show the property is maintained during the execuation of the While loop (i.e., maintenance property).

(e) (10 points) Finally, prove the termination property and conclude.

(f) (10 points) What would happen if $a > 0$ and $b = 0$ and thus the pre-condition is not satisfied?

2. Now, consider another version of the algorithm of Euclid.

Algorithm 2 Euclid2(a,b)

Require: $a, b \geq 0$
Ensure: $a = GCD(a, b)$
  while $b \neq 0$ **do**
    $t \leftarrow b$
    $b \leftarrow a \mod b$
    $a \leftarrow t$
  end while
  return a

We want to estimate its worst case running time using the big-Oh notation.

(a) (10 points) Let x be a integer stored on n bits. How many bits will you need to store $x/2$?

(b) (10 points) We note that if $a \geq b$, then $a \mod b < a/2$. Assume the values of the input integers a and b are encoded on n bits. How many bits will be used to store the values of a and b at the next iteration of the While loop?

(c) (10 points) Deduce from this observation, the maximal number iterations of the While loop the algorithm will do.

(d) (10 points) Assuming the Euclidian division of two integers of n bits is executed in $\mathcal{O}(n^2)$, give a big-Oh notation of the worst-case running time of the algorithm Euclid2.

3. Optional question **(not graded)**: Adapt the proof of the first question to prove the second algorithm.