University of Waterloo
CS240E, Winter 2025
Assignment 0

Due: Tuesday, January 14, 2025 at 5pm, with grace period until 11:59pm

Be sure to read the assignment guidelines (`https://student.cs.uwaterloo.ca/~cs240e/w25/assignments.phtml#guidelines`). For Questions 1 – 6, submit your solutions to Crowdmark. You should also submit the LaTeX-code of your `a0.tex`, `polar_bear.jpg`, and the other image you choose for Q6 to Crowdmark. Don't forget to also submit AID01.

This assignment is worth up to 6 bonus marks, which will be added to your total mark (raw score) for assignment 1.

Introduction

A0 is designed to introduce you to LaTeX. You are strongly encouraged to create all assignment solutions using LaTeX, as it will strongly benefit both you and your markers. While LaTeX is not required for CS240E (except for Q2–6 of this assignment), non-LaTeX submissions are expected to be of comparable presentation and sufficiently messy, hard-to-read solutions[1] may be penalized.

Learning LaTeX is a great asset to have for any course, and also especially for those of you planning to go into academia. As a beginner in LaTeX, like in HTML, it is best to start with an example. (And like HTML, LaTeX comes with considerable legacy; some features are clearly deprecated, but still exist for backward compatibility.)

To complete the problems below, open the LaTeX file used to make this PDF. Inside the file you will find the code used to write this file along with comments explaining the code to help you get through the assignment. If you get stuck there are also many on-line resources you can use; in particular `http://tex.stackexchange.com` is a valuable resource. Searching for "fraction example LaTeX" is acceptable; searching for "LaTeX proof of summation from 1 to n" is **not** acceptable (academic violation).

To compile the .tex file provided simply type "`pdflatex a0.tex`" in the school's Linux environment. LaTeX compilers are also free to download on-line; in particular the free online interface `http://www.overleaf.com` is very popular.

[1] What exactly "hard-to-read" means is at the marker's interpretation—if you want to be sure it is readable, use LaTeX!

0 Academic Integrity Declaration

In order to ensure academic integrity during the term, you must read and sign an Academic Integrity Declaration and submit it before the deadline.

This agreement, which covers A0, A1, A2, and PQ1, will indicate what you must do to ensure the integrity of your grade. **Please note that you need to submit the Academic Integrity Declaration even if you do not plan on doing A0. Failure to do this before the deadline will lead to a grade of 0 for assignments A0, A1, A2, and PQ1.**

1 Assignment Guidelines

At the top of this assignment is the URL to the assignment guidelines for CS240E; it can also be found from the course webpage from the Assignments tab. Please answer the following questions about the assignment guidelines:

a) If an assignment question asks you to design an algorithm, what are the three other things you must do in addition to describing/writing the pseudocode for the algorithm?

b) For programming questions, what programming language do you have to use? To what system should you upload your code?

2 Mathematics

In CS 240E, you will be using many mathematical concepts. It is important to be able to typeset mathematics in your assignments. This will include sums, fractions, subscripts & superscripts, etc. Example:

$$\bar{f}(n) \coloneqq \sqrt{\sum_{i=0}^{\lg n} 4^i \left(\frac{n_0}{2^i}\right)^{\theta}}.$$

In order to practice this skill, write a proof showing:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

Hint: For short formulas, we use inline math surrounded by `$`*: "Let* $n \geq 1$ *be a positive integer." Whitespace is ignored in math mode.*

3 Pseudocode

In CS 240E, you will often need to describe algorithms, for which typically you should give pseudocode. There are many different tools for writing pseudocode in LaTeX; the one we give here is using the package `algorithm2e`.

In the pseudo-code below, there is an error that would make the algorithm crash. Submit a corrected version of the pseudo-code. (Hint: consult the course notes.)

Algorithm 1: *insertion-sort*$(A, n \leftarrow A.size)$

Input : Array A of size at least n

```
1 for i ← 1 to n − 1 do
2     j ← i
3     while j ≥ 0 and A[j] < A[j − 1] do
4         swap A[j] and A[j − 1]
5         j ← j − 1
```

4 Trees

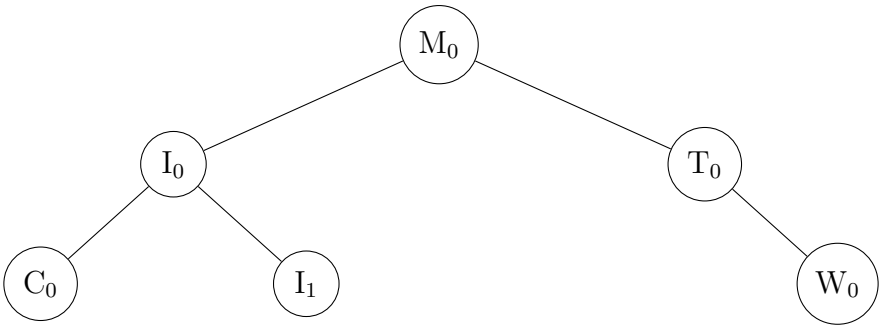
CS 240E introduces many tree data structures. Here is a BST on six letters of the alphabet. Insert the first three letters of your first name into the tree (if your first name is shorter than three letters, simply insert all the letters), starting with the first letter of your name. If you are inserting duplicate letters:

(1) Find the largest index of the letter you are inserting.

(2) Insert your letter, with an index one larger than the index you found.

(3) When comparing to an equal value, break the tie according to the index.

For example, if you were to insert an 'M' into the tree below, it would be entered as M_1 and it would become the left child of T_0. Only show the resulting tree.



Hint: For nodes with only one child, you should use "`child[missing]`" for the non-existent child to keep the binary search tree looking appropriately.

5 Tables

Occasionally, you may want to present information in a table. In LaTeX you can easily present data in well-structured tables. Fill in the table below with any animal you like.

Animal's Name	Avg. Weight	Longevity	Avg. Temperature	Conservation Status
Polar Bear	350-700kg	25 years	37°C	Vulnerable

6 Images

You may find it too time-consuming to do parts of your assignment in LaTeX, at which point you may want to include an image of your work. LaTeX also supports images. Please keep your image sizes small both for this assignment and future assignments, the total file-size should be at most 5MB. However, be sure that your images can be easily read by your markers, or you run the risk of losing marks.

For this question, include an image of the animal you added to the table in Q5 along with a caption (see example below).

Figure 1: Polar Bear.

Hint: (`figure` is a floating environment that gets put where it nicely fits the page layout. The optional argument says which positions are acceptable for the float: top/bottom of a page, here, and on a separate page of floats.)