<div align="center">

St. Francis Xavier University
Department of Computer Science

CSCI 355: Algorithm Design and Analysis
Assignment 1
Due September 19, 2024 at 11:30am

</div>

Assignment Regulations.

- This assignment must be completed individually.

- Please include your full name and email address on your submission.

- You may either handwrite or typeset your submission. If your submission is handwritten, please ensure that the handwriting is neat and legible.

[7 marks] 1. While the Gale–Shapley algorithm is a purely mechanical procedure, humans are humans, and humans tend to find a way to get what they want. Unscrupulous humans may resort to lying in order to get what they want, and this introduces a weakness in the Gale–Shapley algorithm: being untruthful could allow someone to "game" the matching produced by the algorithm.

Suppose we have the following preference lists for hospitals and students:

St. Martha's	Catriona	Aisling	Brody
Aberdeen	Aisling	Catriona	Brody
IWK	Catriona	Aisling	Brody

Aisling	St. Martha's	Aberdeen	IWK
Brody	St. Martha's	Aberdeen	IWK
Catriona	Aberdeen	St. Martha's	IWK
Catriona	*Aberdeen*	*IWK*	*St. Martha's*

In the list on the right, Catriona's true preferences are shown in the third row, while her *untruthful preferences* are shown in *italic text* in the fourth row.

(In the following parts, if you like, you may implement the Gale–Shapley algorithm in the programming language of your choice, but your answer must include a copy of your source code and a verbose listing of each step of your implementation's output.)

(a) Find the matching produced by the Gale–Shapley algorithm if we were to use Catriona's true preferences. Show all your work. What hospital is Catriona matched with?

(b) Find the matching produced by the Gale–Shapley algorithm if we were to use Catriona's untruthful preferences. Show all your work. What hospital is Catriona matched with?

(c) Did Catriona benefit from being untruthful? What does this tell us about the behaviour of the Gale–Shapley algorithm? (You don't need to explain what this tells us about humans, but you may do so if you're feeling philosophical.)

[6 marks] 2. For each of the following blocks of pseudocode, give as tight a bound as possible on the time complexity of the pseudocode using Big-O notation, written in terms of n. You do not need to give a formal proof, but you should give justifications for each of your answers. You may assume that n is a positive integer.

(a)
```
for ( x = n, x > 0, x = x/2 ):
    for ( y = 1, y < n, y = y*2 ):
        for ( z = 0, z < n, z = z+2 ):
            print "Algorithms are awesome"
```

(b)
```
n = 2^m
for ( i = n, i > 0, i = i-1 ):
    for ( j = 1, j < n, j = j*2 ):
        for ( k = 0, k < j, k = k+1 ):
            print "I love CSCI 355"
```

[6 marks]   3. Arrange the following functions in order from slowest growth rate to fastest growth rate. If function $f(n)$ is in position i of your list and function $g(n)$ is in position $i+1$ of your list, then this is equivalent to saying that $f(n) \in O(g(n))$. Give a brief justification for your ordering.

$$
\begin{aligned}
f_1 &= n \cdot 355 \log(\log(n)) & f_4 &= \sqrt{\pi \cdot n} \\
f_2 &= n^{24} & f_5 &= 4^{n^{512}} \\
f_3 &= 1024^n & f_6 &= 6(n/3) + 16(n/8)
\end{aligned}
$$

[6 marks]   4. Let $f(n)$ and $g(n)$ be functions such that $f(n) \in O(g(n))$. For each of the following claims about Big-O notation, if the claim is always true, give a short (1–2 sentence) justification/proof. Otherwise, if the claim is false, give a counterexample consisting of functions $f(n)$ and $g(n)$ that do not satisfy the claim.

(a) $\log_2(f(n)) \in O(\log_2(g(n)))$.

(b) $f(n)^2 \in O(g(n)^2)$.

(c) $2^{f(n)} \in O(2^{g(n)})$.