Quiz 0 COMP 417

Instructions:
This quiz is not for marks, but to help us get to know you and to understand your skills and background. You do not need to write your name or your student number if you don’t want to. The knowledge items are **NOT required** to do well in the course. It’s always OK to say “I don’t know”, but please do try your best to answer.

Don’t be scared off by these items, since they are present not only to gauge your background, but also your interests.

Part 1: Questions about your preferences and goals
1) What is your major and which year are you in your undergraduate studies?

2) Which classes are you taking this semester? Please enter approximate titles, not course numbers.

3) What programming languages do you have practical experience with, and to what comfort level? Place a star next to your favorite language.

4) Are you planning to do your assignments on the Trottier-building machines?

5) What is roughly the CPU model of your laptop/desktop (e.g. i3, i5, i7), or if you don't know it, the age of your machine?

6) Are you comfortable using a Linux distribution (e.g. Ubuntu) **on your own computer**? Note: The Gazebo 3D simulator runs best on an i5 or i7 CPU, particularly if you have a discrete GPU. So, if **a)** you do not have access to such a machine, or **b)** if you cannot install Ubuntu LINUX, we'll need to know early on in the semester to find a solution.

Part 2: Background Knowledge
For each of the following concepts, estimate your level of knowledge and comfort in the range 0-5 where:
0 means you have never heard the word before, while 5 means you could **comfortably work on assignment depending on this knowledge**. In particular, use only 1 or 2 if you have seen the concept, but have forgotten, or never really know, how to use it or what it means.
Recall, the purpose of this quiz is for us to get to know better what your background requirements are and how to match the course to your interests. There are some topics list below which I do not expect any student to know about, but you might surprise me.

Core Computer Science:

1) Loops, conditionals, classes, modularity
2) Data structures: lists, queues, stacks, hash maps/dictionaries, trees, graphs
3) Threads
4) Callbacks, remote procedure calls
5) Serialization
4) Breadth-first and depth-first search
5) Algorithm A* (A –star)
6) Deep learning
7) Convolutional neural networks

Math:

1) Cross product
2) Matrix multiplication
3) Matrix inversion
4) Determinant
5) Gaussian elimination
6) QR decomposition
7) Cholesky decomposition
8) Singular Value Decomposition (SVD)
9) Least squares
10) The concept of Fourier analysis
Probability:
1) Probability density functions in 1D
2) Probability density functions in higher dimensions
3) Cumulative density functions
4) Variance of a random variable
5) Conditional probability
6) Bayes’ rule
7) Maximum likelihood estimation

Calculus/Numerical Optimization:
1) Single-variable derivative
2) Gradient
3) Jacobian matrix
4) Hessian matrix
5) Taylor approximation
6) Finding local minima/maxima
7) Saddle points
8) Gradient descent/ascent

Tools
1) OpenCV
2) numpy
3) ROS (the Robot operating system)
4) Linux/UNIX