

HIT220 - Assignment 1

Lecturer: Yan Zhang

1 August 2024

Individual Work: This assignment is classed as **Individual work** so will follow normal university conventions on plagiarism.

1. You are encouraged to discuss questions, methodology, and ideas with your classmates in HIT220 however your submission must be your own work. i.e. Your own interpretation of the discussions and your own writing.
2. We recommend you do not exchange papers or show your written assignment workings to others.
3. If two submissions have the same answers that will be considered plagiarism. I will give no marks for both versions.

Rubric: All marks are allocated as

1. Half marks for complete correct answers.
2. Half marks for explanations.

Question 1 (1 mark) *An important feature of Object Oriented Programming is using codes as objects. Describe an example of how objects are used in Python programming.*

- Please note that we did not ask for a code. We require a description of how objects are used, not the code itself.

Question 2 (1 mark) *1. A program generates all the combinations of elements of a set and writes the output to a file at a rate of 1100 combinations per second. How long will it take to generate all the combinations of a set with 4 distinct elements?*

2. *What is the Big-O time complexity of the given arithmetic function? Show working.*

```
def Arithmetic(n):  
    a = 0  
    for i in range(1, n//2):  
        for j in range(1, i//4):  
            a += n**i - j  
    return a
```

Question 3 (2 marks) 1. Write pseudocode or code for an algorithm to find the product of a sequence, P , of n integers.

2. What are the time complexity and space complexity of your algorithm? Note that this will depend on your code.

Question 4 (2 marks) 1. Write a recursive function to generate the n th Fibonacci number.

2. What are the time complexity and space complexity of this function?

3. Is this method more efficient than the iterative method?

Question 5 (1 mark) What is the complexity of the queue and dequeue operations on a queue? Please create a data structure for a queue with different complexity.

Question 6 (3 marks) 1. Write **pseudocode** for calculating the Greatest Common Divisor (GCD) of two numbers. There are many ways to do this. Zero marks for code alone.

2. What is the time complexity of this calculation in Big-O notation? This depends on your implementation.

3. Show that the output of this algorithm is indeed the Greatest Common Divisor. Do not simply state that it is; demonstrate why it is.