# Algorithms & Data Structures
## Part 1 Revision Practicals (2024)

These are questions are for you to work on in a practical in Easter Term. After the practicals, I will upload solutions.

After this I suggest you look at the sample exam paper with model solutions that can be found on Sharepoint. You can also work on this in the practical.

You can also find many papers from past years on Sharepoint.

In all cases, I encourage you to attempt questions for yourself before looking at solutions. Simply reading the solutions of others is poor preparation for an exam.

1. (a) What is a stack?

    (b) Give the contents of an initially empty stack after each of the following operations: push(5), push(3), pop(), push(2), push(8), pop(), pop(), push(9), push(1), pop().

    (c) Consider the following code which takes as input a positive integer $n$.

    ```
    stack = Stack()
    While (n > 0) Do
        stack.push(n % 2)
        n = n // 2
    EndWhile

    While (stack.isEmpty == False) Do
        print(stack.pop)
    EndWhile
    ```
    (Note that % and // are the modulus and integer division operators respectively (as in Python).)

        i. What does the code print when $n$ is 14?
        ii. For which value of $n$ does the code print 10011?

2. (a) Describe the result of the following series of operations performed on an empty queue: enqueue(7), enqueue(9), dequeue(), enqueue(4), enqueue(2), dequeue(), dequeue(), enqueue(4).

    (b) A client performs a sequence of push and pop operations on a stack. The push operations put the integers 0 through 9 (in that order) on to the stack; the pop operations print out the return value. For each of the following outputs, describe the sequence of push and pop operations that produced it (or explain why no such sequence exists).
        i. 2 5 6 7 4 8 9 3 1 0
        ii. 1 4 2 9 8 6 5 3 0 7

3. Suppose someone borrows money from a friend and agrees to pay back a constant amount each month. Interest is also added to the amount owed each month. We want to calculate the amount that is still owed at the end of each month.

    For example, suppose the sum borrowed is 1000, the interest rate is 5% and monthly payments of 100 are made. Then after 0 months the amount owed is

    $$A_0 = 1000.$$

    During the first month interest is added ($1000 \times 0.05 = 50$) and a repayment is made so that the amount now owed is
    $$A_1 = 1000 + 50 - 100 = 950.$$

Similarly at the end of the second month the amount outstanding is

$$A_2 = 950 + 950 \times 0.05 - 100 = 897.5.$$

Write a recursive function that takes as input the sum borrowed, the interest rate, the monthly repayment and the number of months that have passed and returns the amount still owed.