STOR-609 Assessment 1 - Marks and Feedback

March 3, 2025

Name : Rebekah Fearnhead

1 Quality of Source Code

- Your implementation is concise and effective.
- You have provided effective commenting making your implementation easy to understand and **re-use**.

Mark (out of 5) : 5

2 Solution

- For both problems you have provided example code that produces a correct solution.
- Your code is designed so that it was easy for me to change the problem and produce further correct solutions (i.e. your code is both **re-usable** and **re-runnable**)

Mark (out of 5) : 5

3 Understanding Design Principles

- You have provided an implementation for a generic backtracking algorithm which is immediately recognisable as such when compared with the pseudo code provided.
- You have provided very extensive and clear descriptions of your methods, design, and solutions to each of the problems.

Notes / Questions

- The manner in which you provide the procedures to backtracking is somewhat unusual. Why not pass the procedures directurly to your backtracking method (higher order programming). This would eliminate the need for your global methods.
- It would be useful for you to consider the computational complexity of your methods in more detail (i.e. how much extra "computation" is required as your problem size increases).

Mark (out of 5) : 4

4 Quality of Written Communication

You have provided very thorough descriptions of how the methods you use for each problem work.

Suggestion

- It would be really good if you also provided pseudo code for each of your **accept**, **reject**, **first**, and **next** methods.

Mark (out of 5) : 4.5

5 Overall Comments and Marks

- You correctly identify the issues associated with stack size and recursion. It is unfortunate that many important algorithms are better expressed recursively (in terms of ease of understanding and analysis) yet many programming systems do not effectively support it. However, there are other programming systems that do support (tail) recursion such as F sharp.
- Do you remember looking at bitwise operations in the **Introductory Python Course** ? How might they be useful in the context of this assignment ?

Overall Mark (out of 20) : 18.5