Practical Task 4.2

(Pass Task)

Submission deadline: Wednesday, April 9
Discussion deadline: Friday, May 2

General Instructions

A Skip-List is a probabilistic Linked List-like data structure that expects, for an ordered sequence of n elements, a $O(\log n)$ runtime complexity for basic operations such as searching, insertion and removal. Importantly, it supports a quick insertion that is not possible for a simple array. Fast operations are made possible by maintaining a linked hierarchy of layered linked-list subsequences, with each successive subsequence skipping over fewer elements than the previous.

1. This task asks you to conduct small research about this data structure. You can find its detailed explanation in Chapter 10.4 of the course book "Data Structures and Algorithms in Java" by Michael T. Goodrich, Irvine Roberto Tamassia, and Michael H. Goldwasser (2014). You may access the book on-line from the reading list application in CloudDeakin available in Content → Reading List → Course Book: Data structures and algorithms in Java. You, of course, may explore and refer to any other resources covering this topic. We expect you to grasp the idea and structure of the skip-list along with standard operations on it. As the results of your study, you must be able to answer the following questions and discuss the related issues:
   - How exactly is the skip-list constructed from layers of linked-lists? What does each of the linked-lists constituting the skip-list store?
   - What is the height of a skip-list? How can we determine the height when adding new elements?
   - What is the runtime complexity for searching, insertion, and removal operations on the skip-list? What does it mean when we say that the complexity of these operations is '***expected***'? What is the difference between an ***expected* bound** on runtime complexity and a ***worst-case* bound**?
   - Compare the skip-list to other data-structures that you already know, e.g., a list collection, singly and doubly linked lists. What are the advantages and disadvantages of the skip-list? Compare the complexities of the basic operations offered by the skip-list to those of the mentioned data structures.
   - How exactly does each of the standard operations work? Here, you should be ready to explain the sequence of actions required to perform searching, insertion, and removal from the skip-list.
2. Solve the following numeric example. Given the sequence of ***coin tosses***, where H stands for ***head*** and T stands for ***tails***:

   T H T T H H T T H T H T H H T H H T T H H T H T T H T T T H T H T H T T T H T H H T T.

   Build a skip-list containing the following values:

   1 40 11 85 86 5 0 8.

   Show the full state of the skip-list after each insertion. Note that *you must explicitly state the meaning you attach* to *heads* and *tails* at the start of your answer. Remember that you must start tossing the coin from left side of the sequence and may not need to use all the tosses to build the skip-list. You may assume as the height resolution policy that we allow a tower to grow as long as heads (or tails) keep getting returned from the given sequence, which emulates a random number generator.

Submission Instructions and Marking Process

To get your task completed, you must finish the following steps strictly on time.

- **Submit** your answers as a single PDF report via OnTrack submission system.
- Once your answers are accepted by your tutor, you will be invited to **proceed with their discussion through a face-to-face interview**. Specifically, you will need to meet with the tutor to demonstrate and discuss your answers in one of the dedicated practical sessions (run online via MS Teams for online students and on-campus for students who selected to join classes at Burwood\Geelong). Please, come prepared so that the class time is used efficiently and fairly for all students in it. Be on time with respect to the specified discussion deadline.
  You will also need to **answer all additional questions** that your tutor may ask you. Questions will cover the lecture notes; so, attending (or watching) the lectures should help you with this **compulsory** discussion part. You should start the discussion as soon as possible as if your answers are wrong, you may have to pass another round, still before the deadline. Use available attempts properly.

Note that we will not accept your solution after **the submission deadline** and will not discuss it after **the discussion deadline**. If you fail one of the deadlines, you fail the task, and this reduces the chance to pass the unit. Unless extended for all students, the deadlines are strict to guarantee smooth and on-time work throughout the unit.

Remember that this is your responsibility to keep track of your progress in the unit that includes checking which tasks have been marked as completed in the OnTrack system by your marking tutor, and which are still to be finalised. When grading your achievements at the end of the unit, we will solely rely on the records of the OnTrack system and feedback provided by your tutor about your overall progress and the quality of your solutions.