ADVANCED WORKBOOK

Tasks for: DLBCSL01-01 – Algorithms, Data Structures, and Programming Languages

Please ensure the validity of these tasks in the course on myCampus.

Note on copyright and plagiarism:

Task 1:

We need to store n distinct key values in a data structure `D`. A key q is of rank k, if `D` has $k-1$ keys less than q. For a user given positive integer k, we would like to report the key of rank k stored in `D`. Describe the reporting algorithm and analyze its complexity if `D` is implemented as:

a) a sorted array
b) a linked list
c) a Heap

Task 2:

a) Design a hierarchy of classes of vehicles based on the following information:
   (i) a vehicle class with the following fields: make, model, and number of wheels.
   (ii) classes: PassengerVehicle, UtilityVehicle, FireEngine, Motorcycle, Cars, Trucks, Sedans, and Convertibles.
b) Describe the contents of each class and appropriate inheritance relations.
c) Describe another set of classes and inheritance hierarchy that could be defined for vehicles.

Task 3:

A retail chain receives sales data on products from its stores in various locations at the end of the month. The data is in an XML file, as `sale` elements with attributes `product` and `price`. You have written a program that inserts these pairs into a data structure that supports search.

a) Discuss the pros and cons of implementing the data structure as a linked list.
b) Discuss the pros and cons of implementing the data structure as a hash table.

In your explanation, you must consider two specific queries you support. Also consider the cases wherein the data are sorted by product name or sales volume.

Task 4:

Consider an SML implementation of the function `f(x, y, z) = (x+y)*z`. The types of the arguments `x`, `y`, and `z` may each be specified as real, specified as int, or not specified at all. Also, the type of the return value may be specified as real, specified as int, or not specified at all. The function call to `f` has three arguments `x`, `y`, `z`, and each may be a real value or an integer value. Give five examples of declarations and function calls that compute valid outputs and five examples that result in errors. Explain the behavior in each case.

Task 5:

Consider the binary recursive variant of the algorithm for computing the Fibonacci number $F(n), n \geq 0$, and prove by induction that it is correct.

Task 6:

Consider the following program written in Haskell, which finds the factors of a number.

```
factors n = [f | f <-[1..n], mod n f == 0]
main = do
print $ factors
```

Consider writing this in an imperative style in C. Using this example, describe how the imperative programming paradigm differs from the functional programming paradigm.