Mark Lawrence A. Mercado

BITS 2C

Creating Searching Algorithms:

A.

1. Serial Search - A serial search is the most fundamental type of search. Prior to starting the search, criteria are established. The search then begins with the first item and goes through each item in turn until it either finds a match or runs out of data without finding one.

2. Binary Search - An effective algorithm for narrowing down a list of things is binary search. In order to reduce the number of potential places to just one, it operates by continually cutting in half the section of the list that could contain the object.

B. Stacks

1. Stack Operations - An abstract data type (ADT) called a stack is used to store data linearly. The only end of a stack via which we can add or remove data is the top of the stack.

A stack may be subjected to a wide variety of stack operations. In this essay, we'll cover most of the operations.

2. Implementation of Stacks as Arrays – Stack can be implemented in 1 dimensional array. However, a stack that is constructed using an array only stores a set number of data values. It's really easy to put this into practice. Simply build a one-dimensional array of a certain size, then add or remove values from the top of that array.

3. Initialize Stack -

* T is the datatype of elements in the stack like int, float
* container is the data structure used to initialize your stack. This is optionally and by default, it will be deque<T> and can be set to other values like vector<T>.
* container\_instance is the instance of container type

4. Empty Stack – Is the stack that has no value stored in it.

5. Full stack – Is when the stack has a certain size that has value in it.

6. Push - This function adds an element to the top of the Stack.

7. Full Stack – The top function returns the topmost element of the stack. You should ensure that there are one or more elements on the stack before calling the top function.

8. Pop - This function removes the topmost element from the stack