**Class Activity 36**

1. **What is data structures and why are they crucial in computer scienc**e?

Data structures are different ways of holding values or information. They are important because each data structure functions in a different way and has different uses.

1. **Take the Scenarios and identify which data structure would be best suited to use**:

[LinkedList, ArrayList, HashMap, Arrays, Binary Search Tree, Queues, Stacks]

* 1. **Task Management System:**
     + Your client would like for you to create a digital system where tasks are cataloged by unique identifiers and frequently accessed or modified by these identifiers, efficiency in search, addition, and removal operations is paramount. The system's design should ensure that accessing any task's details via its unique identifier is extremely quick, suggesting an underlying structure optimized for key-based access.
     + **Which data structure is best suited for this scenario?**

Hashmap

* 1. **Photo Editing Software**:
     + Your client would like you to create a photo editing application that allows users to continuously add new effects to an image and needs to allow them to easily undo recent changes one at a time, or redo them if they change their mind. The order in which effects are applied or removed is critical, implying a *sequential access* mode that enables direct operations at both ends
     + **Which data structure is best suited for this scenario?**

Array

* 1. **Print Job Management**:
     + In an office setting, a printer receives documents to be printed from various computers. It's crucial that these documents are printed in the exact order they're sent to the printer, without any prioritization or reordering, indicating a structure that inherently supports first-in, first-out processing.
     + **Which data structure is best suited for this scenario?**

Queue

* 1. **Web Browser History**:
     + Consider a web browser that tracks the URLs visited by a user in a session to enable navigating backwards and forwards through the history. The most recent visits are the first to be revisited when going ***back***, hinting at a structure that easily allows adding and removing items from the same end.
     + **Which data structure is best suited for this scenario?**

Stacks

* 1. **Product Inventory List**:
     + An online store needs to manage its product listings, where products can be frequently added, removed, or accessed randomly. The need for dynamic resizing and quick access to any product based on its position in the list suggests a flexible structure that can adjust size efficiently while maintaining direct index-based access.
     + **Which data structure is best suited for this scenario?**

ArrayList

* 1. **Game Score**:
     + Your client wants a gaming console to track a fixed number of the highest scores achieved across all users. The scores need to be updated every time a new score makes it into the top list, but the total number of scores tracked remains constant, indicating a structure that is optimal for fixed-size collections and quick, direct access.
     + **Which data structure is best suited for this scenario?**

Binary Search Tree

* 1. **User Contact Information**:
     + An application manages a large, dynamic set of user contact details that must be kept in a sortable order, with frequent insertions and deletions as users update their information. The system is designed to facilitate quick searches and efficient updates, maintaining order at all times, suggesting a self-balancing approach to handling data.
     + **Which data structure is best suited for this scenario?**

LinkedList

1. **List and describe the 4 principles of Object-Oriented Programming, give examples for each one of when we used each of them throughout the semester**.

Abstraction: is the process of hiding certain details and only showing essential details or  
functionality to the user. The non-essential details are not displayed. We used this in project 3 and similar labs.

Encapsulation: refers to the bundling of data (variables) and methods that operate on the  
data within a single unit. We used this in some activities like 16.

Inheritance: s a way to reuse parts of an existing class inside of new classes. The new  
classes inherit or acquire the properties (variables and methods) of the existing class. It is a way to create new classes based on existing classes. We did a inheritance activity with flowers.

Polymorphism: is when the same method call can lead to different behaviors depending on the type of object on which the method call is made. A single action can be performed in many ways. We have done this with some of our recent projects and labs when we created more instances of other classes.