**CMPSC-462 –Data Structures**

**PROJECT-1**

**Stack and Queue Applications**

In this project, you will design your own class implementing Tower of Hanoi, Palindrome detector and another application of your interest using stacks and queues.

This project will give you more experience on the use of:

1. stack and queue implementations

2. Class, class methods

**Project Specifications:**

**Part-1**

1. Implement a class, named Stack, for a standard stack using a List data structure. The stack should have the following methods:
2. push(item)
3. pop()
4. peek()
5. isEmpty()
6. size()

2. Create a class Queue that implements a queue data structure. It should have the following methods:

1. enqueue(item)
2. dequeue()
3. isEmpty()
4. size()
5. Create a class Deque that implements a deque data structure. Show an example implementing Deque operations.

**Part-2: Applications**

1. Tower of Hanoi is a simple game which is usually used to demonstrate the use of recursion in algorithms. **Implement the game using**
2. **recursive functions and**
3. **iterative functions.** We are going to solve the game using stacks and queues.

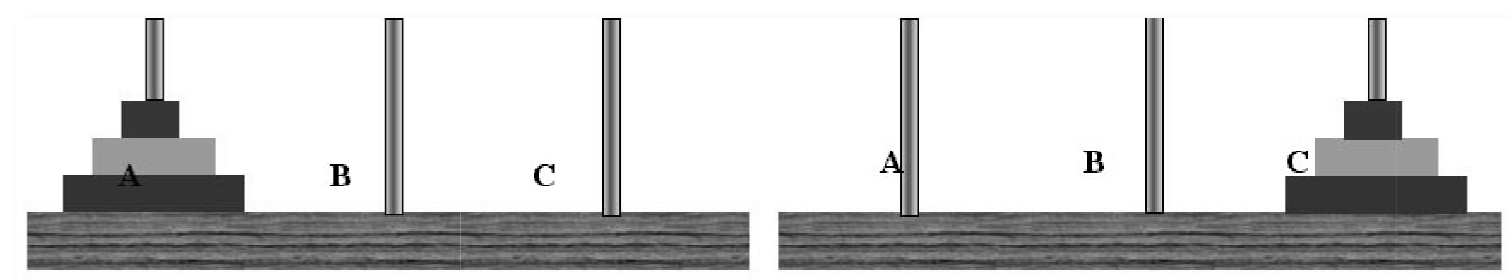
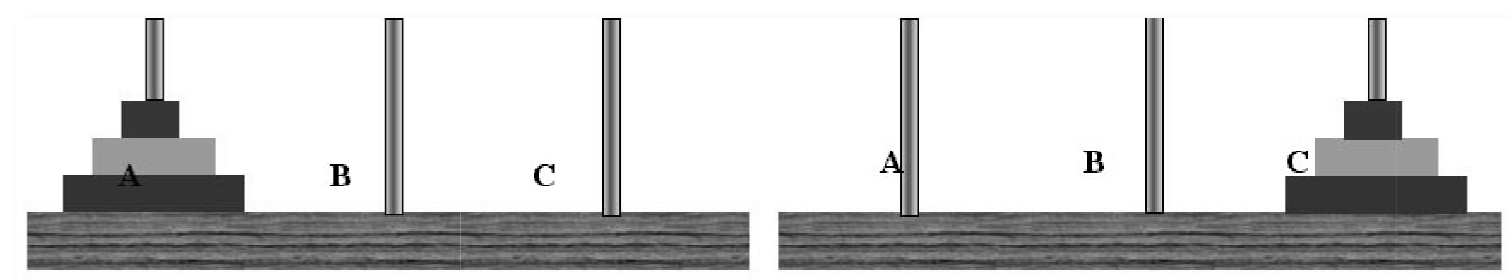
 

Figure-a Figure-b

The objective of the game is to move the stack of disks (as shown in Figure-a) to a rod on the extreme right (as shown in Figure-b) following the below mentioned rules:

* Only one disk can be moved at a time
* A disk cannot be placed over a smaller disk

**Implementation using Stack:**

Develop a class StackTower to implement Tower of Hanoi

* You have to create three stack objects representing the rods
* You need to push minimum three disk objects onto the rods to demonstrate the simulation
* Then you will pop from one rod and push it onto another rod
* Repeat this process while making sure that you are not placing a disk over a smaller one

Your program should print out the contents of the stack at the beginning. After each move print out the contents of all the 3 stacks until the final step where the third rod is populated with all the disks. Show the simulation of solving Tower of Hanoi.

1. Consider a real-life situation. Formulate a question and then design a simulation that can help to answer it. Possible situations include:

* Cars lined up at a car wash
* Customers at a grocery store check-out
* Airplanes taking off and landing on a runway
* A bank teller

Choose a stack or queue or both to demonstrate the simulation. Be sure to state any assumptions that you make and provide any probabilistic data that must be considered as part of the scenario.

1. Choose another application of your interest to demonstrate Deque abstract data structure mechanism.

**Deliverables:**

1. a single .py file containing the project.
2. a report - a word file including the following:

* Outputs with appropriate screenshots.
* Algorithms used for each application and brief explanation.
* Conclusion paragraph. Conclusion should also address:
* What have you learnt from this project?
* What ways you can expand this project?

/\* All students are expected to use appropriate amount of comments to explain their program. \*/

**Due Date:**

9/22/2022 (9:00 am)

You should use the sample report format attached here for the full report submission.



CMPSC 462: Data Structures (Fall 2020)

**Project-1**

Stack, Queue and Deque Applications

Your name:

Instructor:

Submitted On: Date here

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| 1 | INTRODUCTION | x |
| 2 | BACKGROUND (optional) | x |
| 3 | DESIGN & IMPLEMENTATION | x |
| 4 | RESULTS / SAMPLE OUTPUTS | x |
| 5 | CONCLUSION | x |
| 6 | REFERENCES | x |

1. **INTRODUCTION**

Introduction for the project

// Insert page numbers in all page and so you can refer it in the Outline.

1. **BACKGROUND (optional)**

You can discuss the theory behind any specific functions used in the software

1. **DESIGN & IMPLEMENTATION**

You can also discuss the algorithms here

1. **RESULTS / SAMPLE OUTPUTS**

Note: If you use a large set of data, you can show a part of the screenshot for your sample result.

1. **CONCLUSION**

* What have you learnt from this project?
* What ways you can expand this project?

1. **REFERENCES**

Cite your source of reference here