Tractivity (Activity Tracker)

Submitted by:

Niznik Marie C. Alamban

John Webert Fabria

Eann Marten E. Lao

Deborah Faith M. Rizon

Section:

CCA

Submitted to:

Sir Jessie Lagrosas

Table of Contents

Introduction…………………………………………………………………………….. Page 3

Overview

Objectives

Scope and Limitation

Functionalities

Program Design and Implementation……………………………………………….. Page 3-4

Pseudocode

Data Structures and Algorithms Discussion with Code Snippets

Conclusion……………………………………………………………………………… Page 4

References……………………………………………………………………………… Page 4

Appendices…………………………………………………………………………….. Page 5

Project Proposal

Photo Documentation / Program Screenshots

Tractivity (Activity Tracker)

Niznik Marie C. Alamban
Department of Information Technology
Xavier University - Ateneo de Cagayan
20180013480@my.xu.edu.ph

Eann Marten E. Lao
Department of Computer Science
Xavier University - Ateneo de Cagayan
20180014313@my.xu.edu.ph

John Webert Fabria
Department of Information Technology
Xavier University - Ateneo de Cagayan
20180013951@my.xu.edu.ph

Deborah Faith M. Rizon
Department of Computer Science
Xavier University - Ateneo de Cagayan
20180013241@my.xu.edu.ph

ABSTRACT

People nowadays are having a hard time organizing their schedules and activities, especially students and professionals due to the overwhelming amount of tasks that they need to do. But through this program, they can now organize and keep track of their schedules and activities with ease. It serves as a planner or a schedule organizer. It aims to lessen the hassle that students and professionals experience in organizing their schedules and activities which is why it offers some features which will enable the users to manage their activities efficiently.

CCS Concepts

• CCS → Applied computing → Computers in other domains → Personal computers and PC applications

Keywords

Planner; Schedule Organizer

1. INTRODUCTION

1.1 Overview

The program is capable of allowing users to input their activities so that they can easily keep track of them. Furthermore, the program uses arrays, queues, and file systems so that it can effectively perform its purpose of enabling the people to plan or organize their activities in a more efficient way.

1.2 Objectives

The Activity Tracker aims to assist the students and professionals in organizing their schedules and activities, this way they can easily keep track of their schedules and activities without any inconvenience.

1.3 Scope and limitation

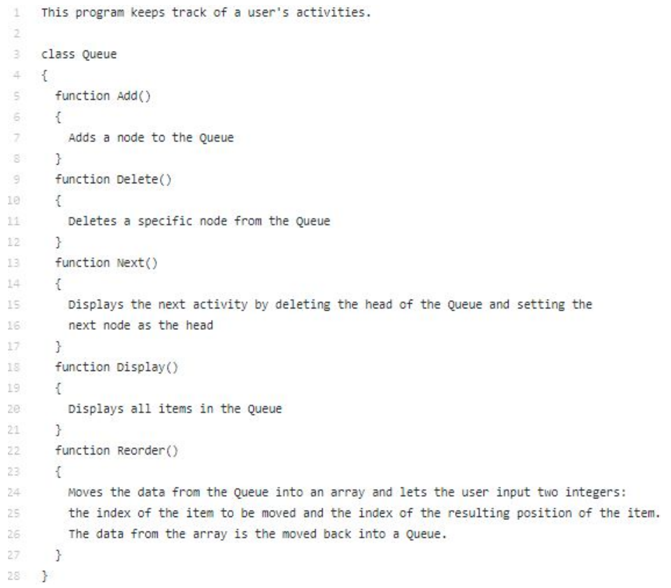
The program is capable of adding, deleting, displaying, and reordering the activities which the user has inputted. Moreover, the program uses arrays, queues and file system. However, it also has its own limitations. The program just simply outputs the activities inputted by the user without the time and date.

1.4 Functionalities

The program allows a user to input homeworks, projects, and other activities. It will then display one of the user's activities at a time. Aside from this, the user also has an option to add, delete, reorder, or display an activity. Moreover, queues, arrays, and file systems are also implemented in this program.

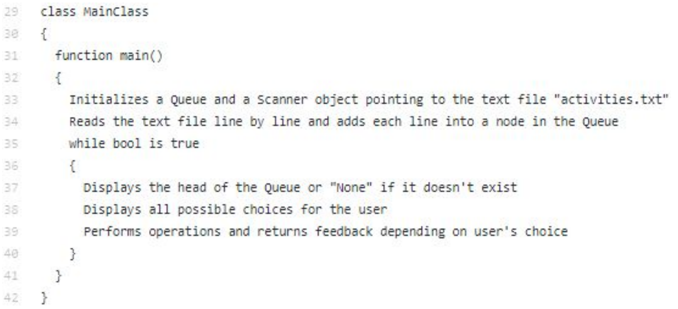
2. PROGRAM DESIGN AND IMPLEMENTATION

2.1 Pseudocode



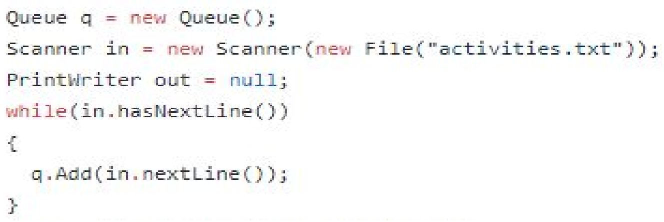
```
This program keeps track of a user's activities.

class Queue
{
  function Add()
  {
    Adds a node to the Queue
  }
  function Delete()
  {
    Deletes a specific node from the Queue
  }
  function Next()
  {
    Displays the next activity by deleting the head of the Queue and setting the
    next node as the head
  }
  function Display()
  {
    Displays all items in the Queue
  }
  function Reorder()
  {
    Moves the data from the Queue into an array and lets the user input two integers:
    the index of the item to be moved and the index of the resulting position of the item.
    The data from the array is the moved back into a Queue.
  }
}
```



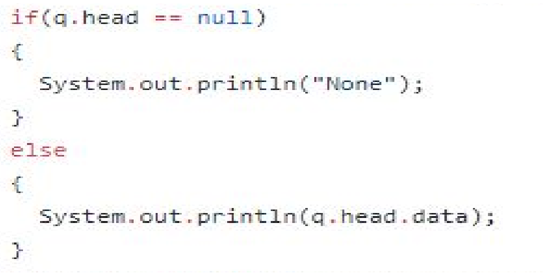
```
class MainClass
{
  function main()
  {
    Initializes a Queue and a Scanner object pointing to the text file "activities.txt"
    Reads the text file line by line and adds each line into a node in the Queue
    while bool is true
    {
      Displays the head of the Queue or "None" if it doesn't exist
      Displays all possible choices for the user
      Performs operations and returns feedback depending on user's choice
    }
  }
}
```

2.2 Data Structures and Algorithms Discussion



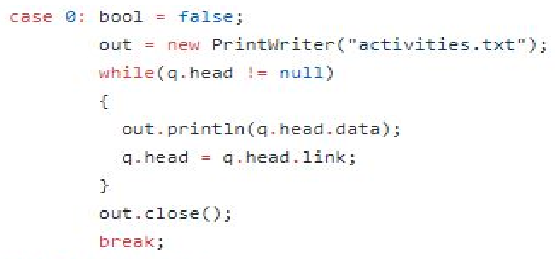
```
Queue q = new Queue();
Scanner in = new Scanner(new File("activities.txt"));
PrintWriter out = null;
while(in.hasNextLine())
{
  q.Add(in.nextLine());
}
```

1) This code takes data from the text file.



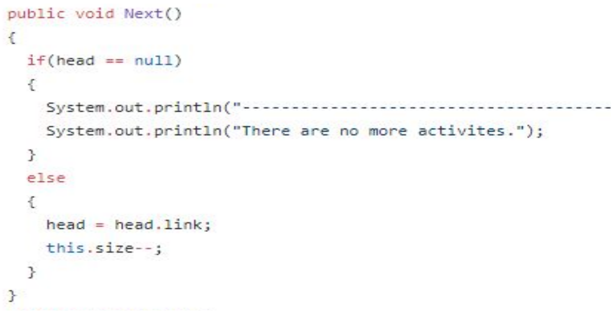
```
if(q.head == null)
{
  System.out.println("None");
}
else
{
  System.out.println(q.head.data);
}
```

2) Displays the current activity.



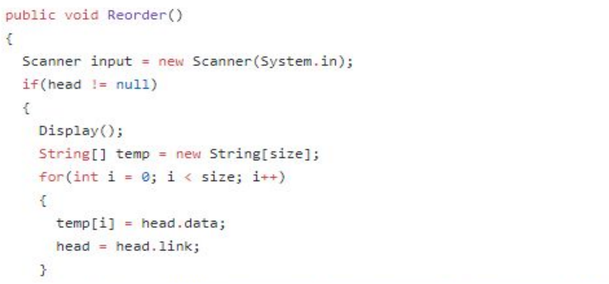
```
case 0: bool = false;
        out = new PrintWriter("activities.txt");
        while(q.head != null)
        {
          out.println(q.head.data);
          q.head = q.head.link;
        }
        out.close();
        break;
```

3) Writes to the text file.

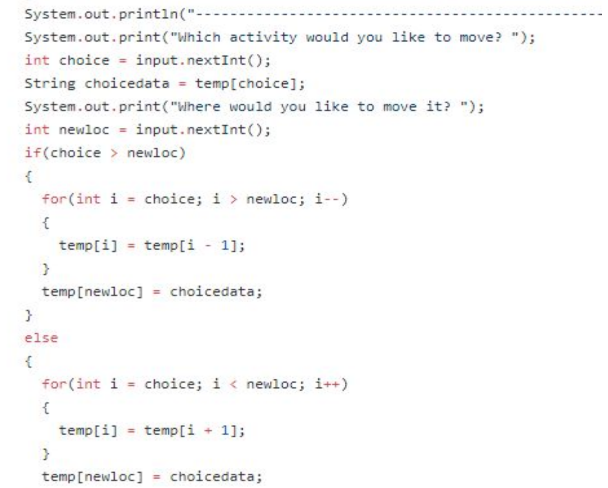


```
public void Next()
{
  if(head == null)
  {
    System.out.println("-------------------------------------
    System.out.println("There are no more activites.");
  }
  else
  {
    head = head.link;
    this.size--;
  }
}
```

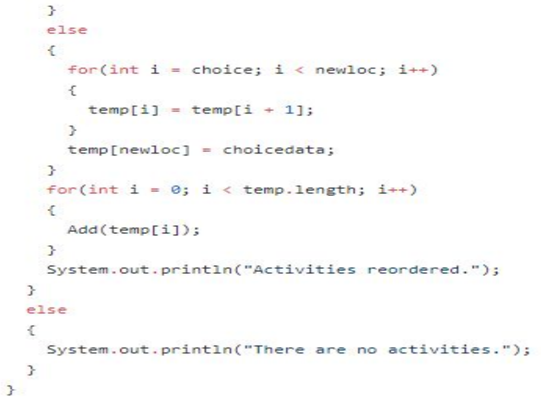
4) Deletes current activity and moves to the next.



```
public void Reorder()
{
  Scanner input = new Scanner(System.in);
  if(head != null)
  {
    Display();
    String[] temp = new String[size];
    for(int i = 0; i < size; i++)
    {
      temp[i] = head.data;
      head = head.link;
    }
```



```
System.out.println("-----------------------------------------------
System.out.print("Which activity would you like to move? ");
int choice = input.nextInt();
String choicedata = temp[choice];
System.out.print("Where would you like to move it? ");
int newloc = input.nextInt();
if(choice > newloc)
{
  for(int i = choice; i > newloc; i--)
  {
    temp[i] = temp[i - 1];
  }
  temp[newloc] = choicedata;
}
else
{
  for(int i = choice; i < newloc; i++)
  {
    temp[i] = temp[i + 1];
  }
  temp[newloc] = choicedata;
```



```
  }
  else
  {
    for(int i = choice; i < newloc; i++)
    {
      temp[i] = temp[i + 1];
    }
    temp[newloc] = choicedata;
  }
  for(int i = 0; i < temp.length; i++)
  {
    Add(temp[i]);
  }
  System.out.println("Activities reordered.");
}
else
{
  System.out.println("There are no activities.");
}
}
```

5) Reorders the activities.

3. CONCLUSION

With what is done for the program, it can be concluded that this activity tracker can aid students and even professionals to be well-organized with their schedules. This can make a student's academic life much easier in a way that it will help them understand their schedule since it is already organized and overall, it can help reduce the amount of stress that they are experiencing just by organizing their schedules. Although the program has its own features which are great in aiding students or professionals, the activity tracker also has its own limitations. In the future, a lot of improvements can still be made because as of now, the program is still simple. It can be enhanced by adding more features so that the users can have more options in organizing their activities and also for them to have a great user experience. But despite of its simple features, the program is good enough for one student to understand how it functions. In short, the program is functional even in its simplistic form.

4. REFERENCES

[1] Juneau, J. 2014. Java 8 Recipes. Books 24x7.

[2] 2017. Different ways of Reading a text file in Java. GeeksforGeeks.

5. APPENDICES

5.1 Project Proposal

Title of the Program: Activity Tracker

Brief Description of the Program:

A life of a student can become hectic when they have to organize their schedule and activities. But with this certain program, the *Activity Tracker*, it will help by allowing users to manage in a better way on scheduling their activities. This program serves as a planner or a schedule organizer. Now students, and also professionals, can easily organize and keep track of their homeworks with ease, and use different other features the Activity Tracker can offer to them for their academic life.

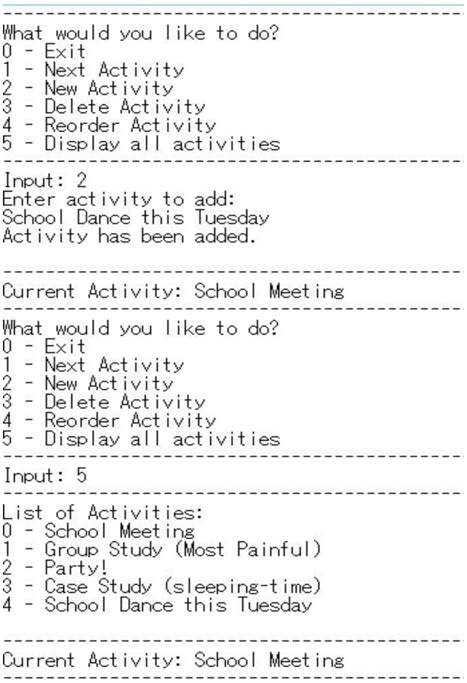
Functionalities and Features:

The purpose of this program is to aid students and even non-students on scheduling their daily activities. This is a great way for them to easily keep track on the assignments and activities that they need to do. The user will simply input his or her homeworks, activities, and more on the program then it will display one of the user's activities at a time. There are also options for the user to rearrange, delete, or go to the next activity. Furthermore, this program is an application of the topics on queues and file system.

Possible data structures and algorithms to be used:

- Queues
- Linked Lists
- Arrays

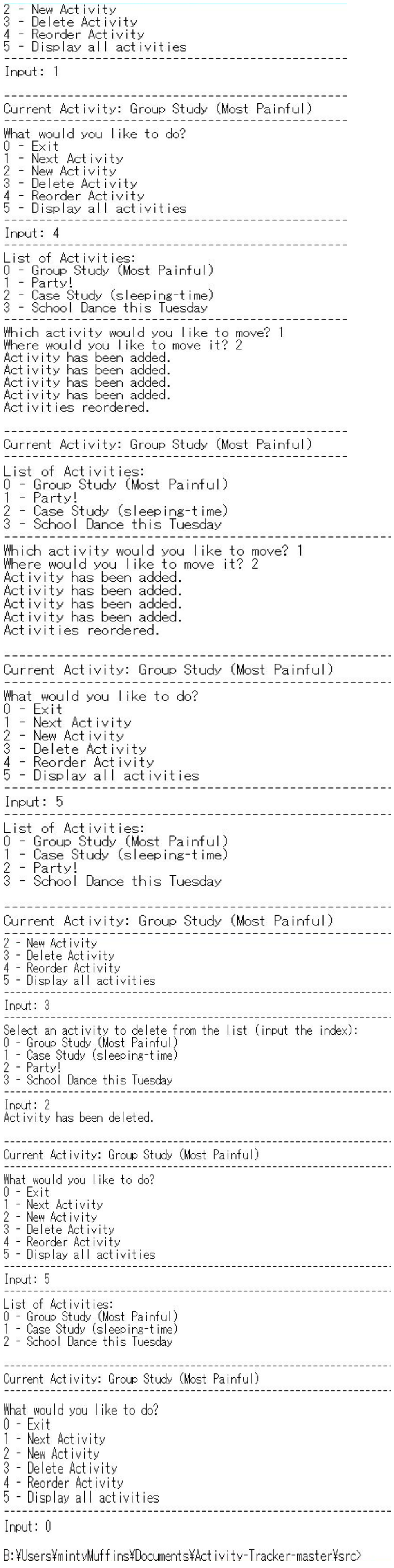
5.2 Program Screenshots



```
-------------------------------------------
What would you like to do?
0 - Exit
1 - Next Activity
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 2
Enter activity to add:
School Dance this Tuesday
Activity has been added.

-------------------------------------------
Current Activity: School Meeting
-------------------------------------------
What would you like to do?
0 - Exit
1 - Next Activity
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 5
-------------------------------------------
List of Activities:
0 - School Meeting
1 - Group Study (Most Painful)
2 - Party!
3 - Case Study (sleeping-time)
4 - School Dance this Tuesday

-------------------------------------------
Current Activity: School Meeting
-------------------------------------------
```



```
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 1

-------------------------------------------
Current Activity: Group Study (Most Painful)
-------------------------------------------
What would you like to do?
0 - Exit
1 - Next Activity
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 4
-------------------------------------------
List of Activities:
0 - Group Study (Most Painful)
1 - Party!
2 - Case Study (sleeping-time)
3 - School Dance this Tuesday
-------------------------------------------
Which activity would you like to move? 1
Where would you like to move it? 2
Activity has been added.
Activity has been added.
Activity has been added.
Activity has been added.
Activities reordered.

-------------------------------------------
Current Activity: Group Study (Most Painful)
-------------------------------------------
List of Activities:
0 - Group Study (Most Painful)
1 - Party!
2 - Case Study (sleeping-time)
3 - School Dance this Tuesday
-------------------------------------------
Which activity would you like to move? 1
Where would you like to move it? 2
Activity has been added.
Activity has been added.
Activity has been added.
Activity has been added.
Activities reordered.

-------------------------------------------
Current Activity: Group Study (Most Painful)
-------------------------------------------
What would you like to do?
0 - Exit
1 - Next Activity
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 5
-------------------------------------------
List of Activities:
0 - Group Study (Most Painful)
1 - Case Study (sleeping-time)
2 - Party!
3 - School Dance this Tuesday

-------------------------------------------
Current Activity: Group Study (Most Painful)
-------------------------------------------
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 3
-------------------------------------------
Select an activity to delete from the list (input the index):
0 - Group Study (Most Painful)
1 - Case Study (sleeping-time)
2 - Party!
3 - School Dance this Tuesday
-------------------------------------------
Input: 2
Activity has been deleted.

-------------------------------------------
Current Activity: Group Study (Most Painful)
-------------------------------------------
What would you like to do?
0 - Exit
1 - Next Activity
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 5
-------------------------------------------
List of Activities:
0 - Group Study (Most Painful)
1 - Case Study (sleeping-time)
2 - School Dance this Tuesday

-------------------------------------------
Current Activity: Group Study (Most Painful)
-------------------------------------------
What would you like to do?
0 - Exit
1 - Next Activity
2 - New Activity
3 - Delete Activity
4 - Reorder Activity
5 - Display all activities
-------------------------------------------
Input: 0

B:¥Users¥mintyMuffins¥Documents¥Activity-Tracker-master¥src>
```