Table of Contents

CAB301 PROJECT REPORT

INTRODUCTION

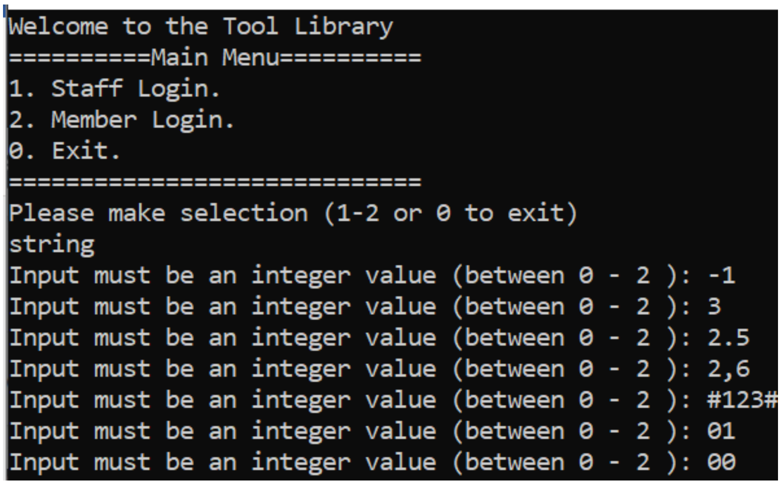
This project works on a library system of tools that has 9 main categories and each has unique tool types for members to borrow. All the transactions/actions/updates from "staff" and "member" are systematically recorded by the application. By using the given criteria and detailed design scenario, I have implemented all needed interfaces and made use of most of the given functions. Furthermore, a test plan of the application and a detailed analysis of the algorithm has been finished which are demonstrated below.

TEST PLAN

Incorrect user selection: (input not in the menu)

*__note__: this check user's inputs method applied to all menu in the program that requires user input to demonstrate next interface. (ex: Tool's category menu, staff menu, member menu, tool's type menu, tool option menu, member list menu, member's borrowing tool list menu, etc.)

Outcome: (take main menu as an example)

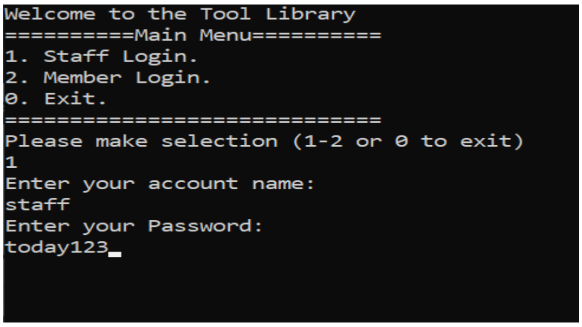


```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
string
Input must be an integer value (between 0 - 2 ): -1
Input must be an integer value (between 0 - 2 ): 3
Input must be an integer value (between 0 - 2 ): 2.5
Input must be an integer value (between 0 - 2 ): 2,6
Input must be an integer value (between 0 - 2 ): #123#
Input must be an integer value (between 0 - 2 ): 01
Input must be an integer value (between 0 - 2 ): 00
```

Main menu functions:

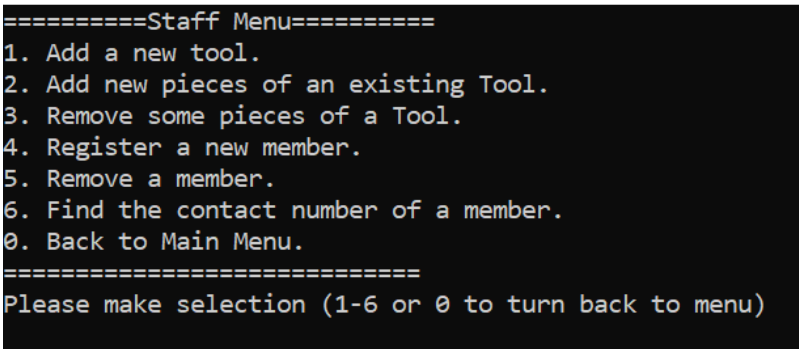
1) Login as staff:

a) **Correct staff login:** using account name ”staff”, password: “today123”



```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
1
Enter your account name:
staff
Enter your Password:
today123_
```

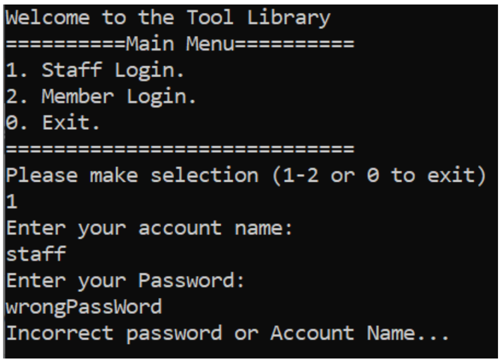
Outcome: staff menu



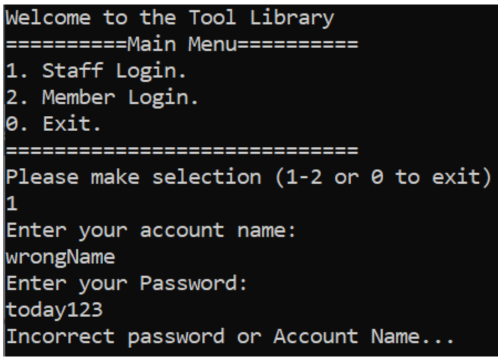
```
==========Staff Menu==========
1. Add a new tool.
2. Add new pieces of an existing Tool.
3. Remove some pieces of a Tool.
4. Register a new member.
5. Remove a member.
6. Find the contact number of a member.
0. Back to Main Menu.
=============================
Please make selection (1-6 or 0 to turn back to menu)
```

b) **Wrong staff login**: use wrong account name or password.

Outcome:



```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
1
Enter your account name:
staff
Enter your Password:
wrongPassWord
Incorrect password or Account Name...
```

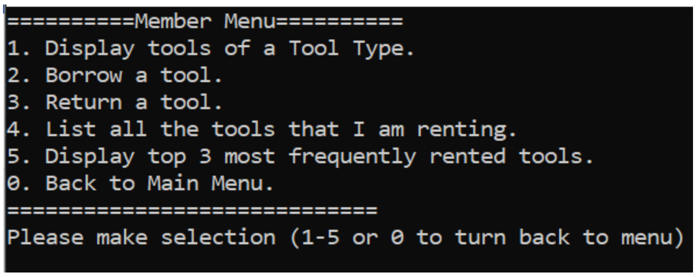


```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
1
Enter your account name:
wrongName
Enter your Password:
today123
Incorrect password or Account Name...
```

2) Member Login

a) Success member login: input correct username and password (and the account must be existed)

- **Outcome**: displays member menu

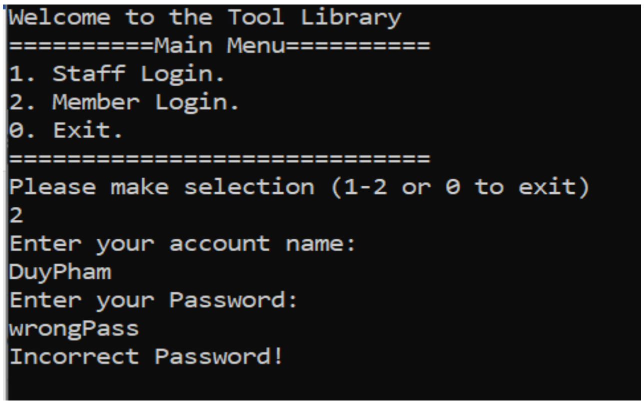


```
==========Member Menu==========
1. Display tools of a Tool Type.
2. Borrow a tool.
3. Return a tool.
4. List all the tools that I am renting.
5. Display top 3 most frequently rented tools.
0. Back to Main Menu.
==============================
Please make selection (1-5 or 0 to turn back to menu)
```

b) Fail to login as a member:

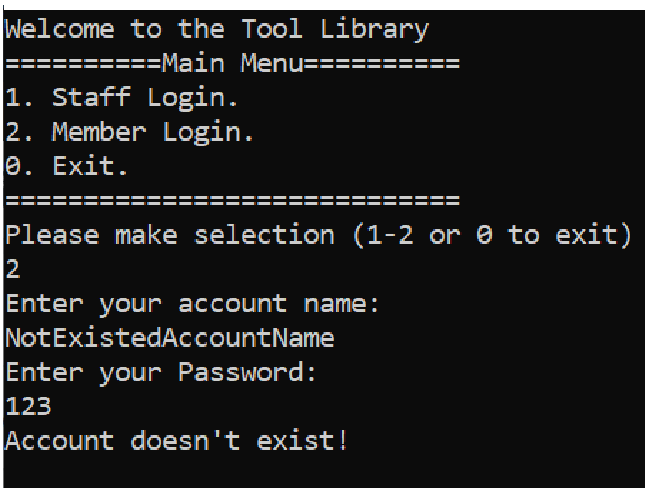
Outcomes:

- Case: input wrong password(existed account)



```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
2
Enter your account name:
DuyPham
Enter your Password:
wrongPass
Incorrect Password!
```

- Case: Input account that is not existed in the system

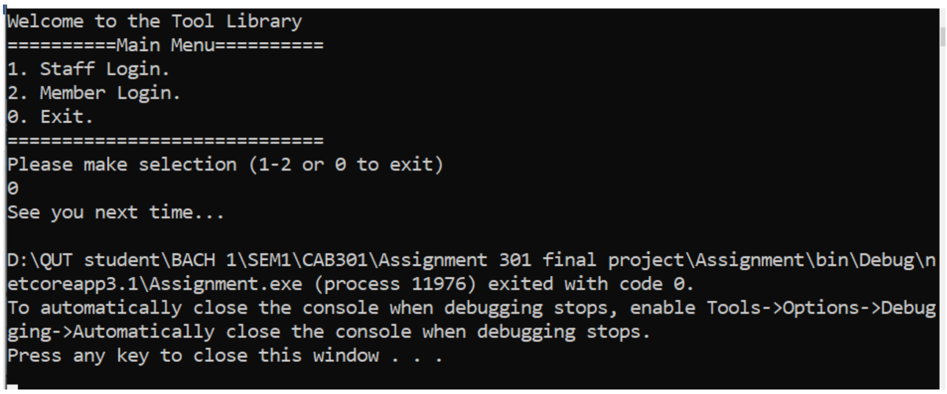


```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
2
Enter your account name:
NotExistedAccountName
Enter your Password:
123
Account doesn't exist!
```

3) Exit program.

a) press “0” to exit the program instantly.

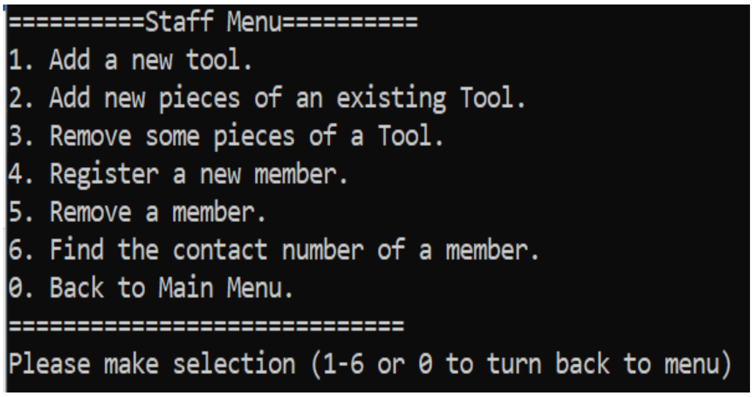
Outcome:



```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
0
See you next time...

D:\QUT student\BACH 1\SEM1\CAB301\Assignment 301 final project\Assignment\bin\Debug\n
etcoreapp3.1\Assignment.exe (process 11976) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debug
ging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

Staff menu functions

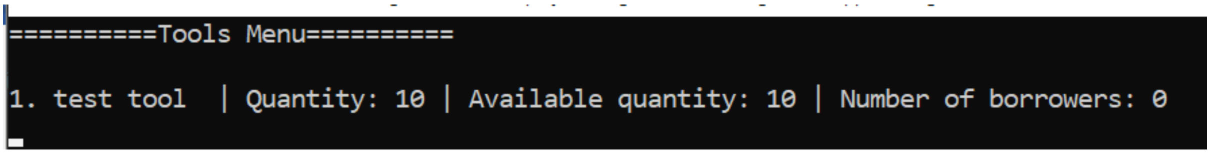


```
==========Staff Menu==========
1. Add a new tool.
2. Add new pieces of an existing Tool.
3. Remove some pieces of a Tool.
4. Register a new member.
5. Remove a member.
6. Find the contact number of a member.
0. Back to Main Menu.
==============================
Please make selection (1-6 or 0 to turn back to menu)
```

1) Add a new tool:

a) Successfully added: (valid quantity input)

Outcome:

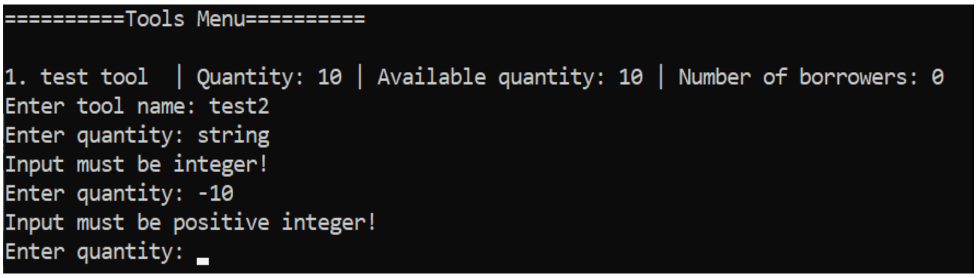


```
==========Tools Menu==========

1. test tool  | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
```

b) Fail to add (invalid quantity input):

Outcome:

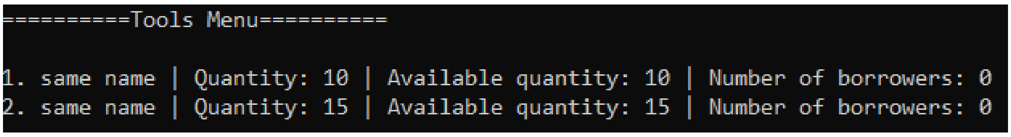


```
==========Tools Menu==========

1. test tool  | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Enter tool name: test2
Enter quantity: string
Input must be integer!
Enter quantity: -10
Input must be positive integer!
Enter quantity:
```

c) Successfully added 2 tools with the same name:

Outcome:



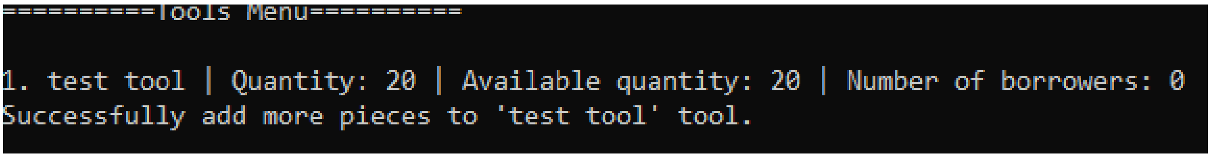
```
==========Tools Menu==========

1. same name | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
2. same name | Quantity: 15 | Available quantity: 15 | Number of borrowers: 0
```

2) Add new pieces of an existing tool:

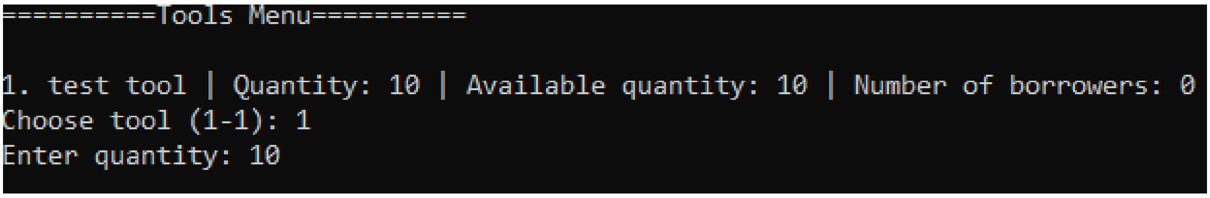
a) Successfully added more pieces. (the tool must be existed to add)

Outcome:



```
==========Tools Menu==========

1. test tool | Quantity: 20 | Available quantity: 20 | Number of borrowers: 0
Successfully add more pieces to 'test tool' tool.
```



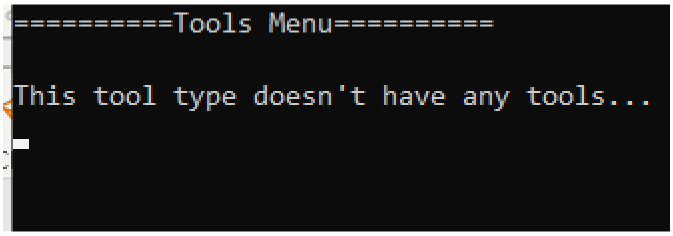
```
==========Tools Menu==========

1. test tool | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-1): 1
Enter quantity: 10
```

b) Fail to add more pieces.

Outcomes:

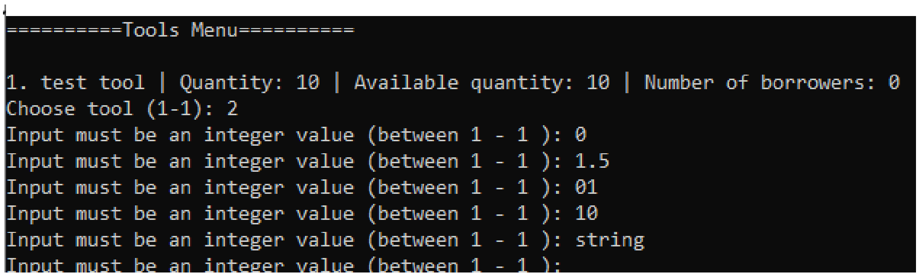
Case: No tool available to add more pieces:



```
==========Tools Menu==========

This tool type doesn't have any tools...
```

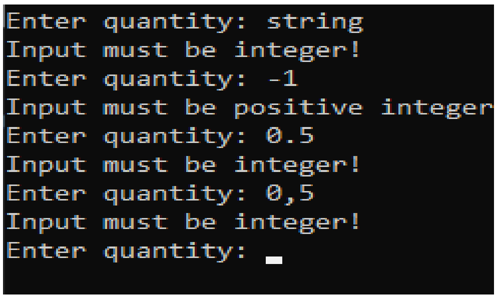
Case: Wrong tool option



```
==========Tools Menu==========

1. test tool | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-1): 2
Input must be an integer value (between 1 - 1 ): 0
Input must be an integer value (between 1 - 1 ): 1.5
Input must be an integer value (between 1 - 1 ): 01
Input must be an integer value (between 1 - 1 ): 10
Input must be an integer value (between 1 - 1 ): string
Input must be an integer value (between 1 - 1 ):
```

Case: Invalid quantity input

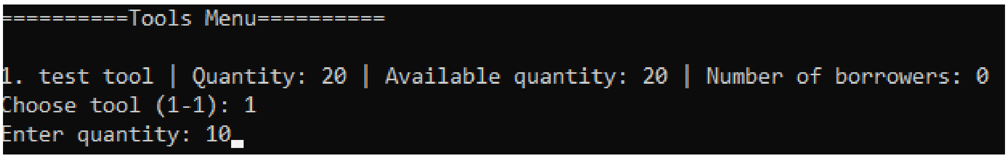


```
Enter quantity: string
Input must be integer!
Enter quantity: -1
Input must be positive integer
Enter quantity: 0.5
Input must be integer!
Enter quantity: 0,5
Input must be integer!
Enter quantity:
```

3) Remove some pieces of a tool:

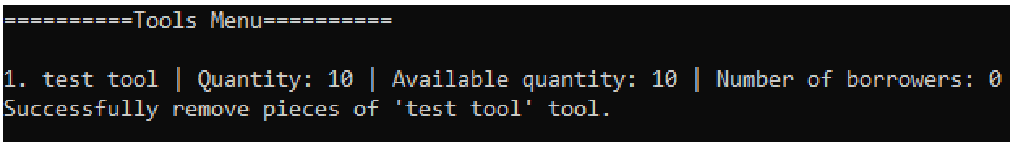
a) Successfully remove pieces:

Outcome:



```
==========Tools Menu==========

1. test tool | Quantity: 20 | Available quantity: 20 | Number of borrowers: 0
Choose tool (1-1): 1
Enter quantity: 10
```



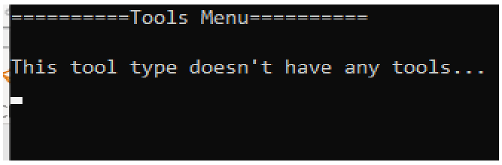
```
==========Tools Menu==========

1. test tool | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Successfully remove pieces of 'test tool' tool.
```

b) Fail to remove pieces:

Outcomes:

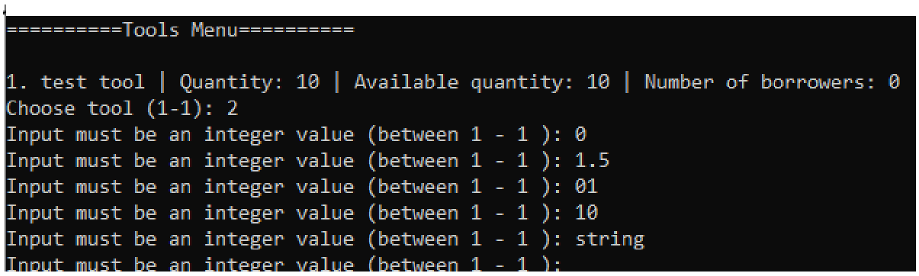
Case: No tools available



```
==========Tools Menu==========

This tool type doesn't have any tools...
```

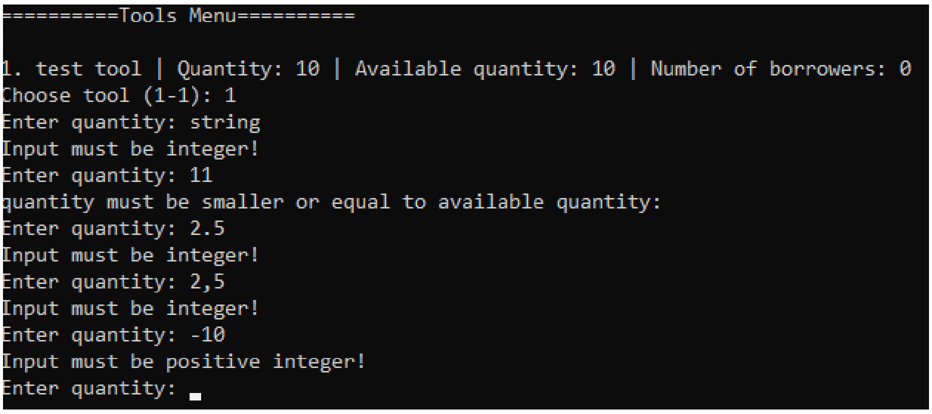
Case: wrong tool option



```
==========Tools Menu==========

1. test tool | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-1): 2
Input must be an integer value (between 1 - 1 ): 0
Input must be an integer value (between 1 - 1 ): 1.5
Input must be an integer value (between 1 - 1 ): 01
Input must be an integer value (between 1 - 1 ): 10
Input must be an integer value (between 1 - 1 ): string
Input must be an integer value (between 1 - 1 ):
```

Case: wrong quantity input



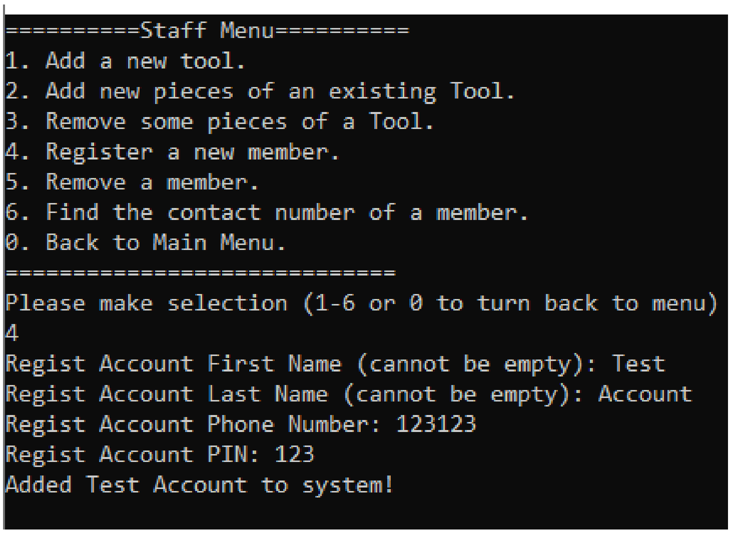
```
==========Tools Menu==========

1. test tool | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-1): 1
Enter quantity: string
Input must be integer!
Enter quantity: 11
quantity must be smaller or equal to available quantity:
Enter quantity: 2.5
Input must be integer!
Enter quantity: 2,5
Input must be integer!
Enter quantity: -10
Input must be positive integer!
Enter quantity:
```

4) Register a new member:

a) Successfully added a new member. (provide valid data)

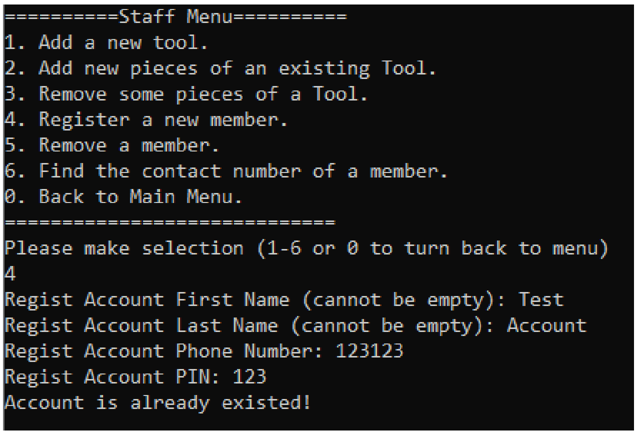
Outcome:



```
==========Staff Menu==========
1. Add a new tool.
2. Add new pieces of an existing Tool.
3. Remove some pieces of a Tool.
4. Register a new member.
5. Remove a member.
6. Find the contact number of a member.
0. Back to Main Menu.
==============================
Please make selection (1-6 or 0 to turn back to menu)
4
Regist Account First Name (cannot be empty): Test
Regist Account Last Name (cannot be empty): Account
Regist Account Phone Number: 123123
Regist Account PIN: 123
Added Test Account to system!
```

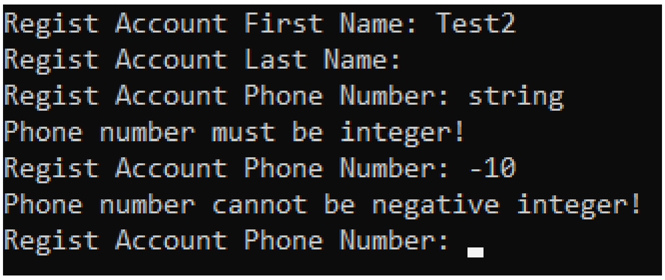
b) Fail to add new member.
Outcomes:

Case: Account already existed.



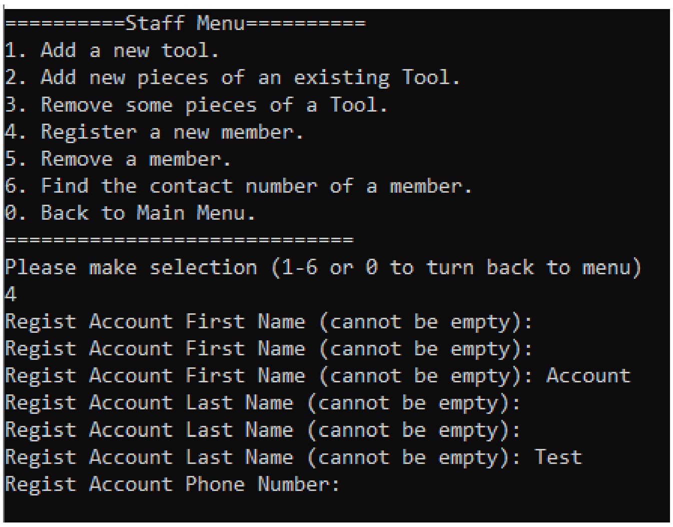
```
==========Staff Menu==========
1. Add a new tool.
2. Add new pieces of an existing Tool.
3. Remove some pieces of a Tool.
4. Register a new member.
5. Remove a member.
6. Find the contact number of a member.
0. Back to Main Menu.
==============================
Please make selection (1-6 or 0 to turn back to menu)
4
Regist Account First Name (cannot be empty): Test
Regist Account Last Name (cannot be empty): Account
Regist Account Phone Number: 123123
Regist Account PIN: 123
Account is already existed!
```

Case: Wrong phone number input.



```
Regist Account First Name: Test2
Regist Account Last Name:
Regist Account Phone Number: string
Phone number must be integer!
Regist Account Phone Number: -10
Phone number cannot be negative integer!
Regist Account Phone Number:
```

Case: Account Name first name or last name is empty (will repeatedly ask until got the input)

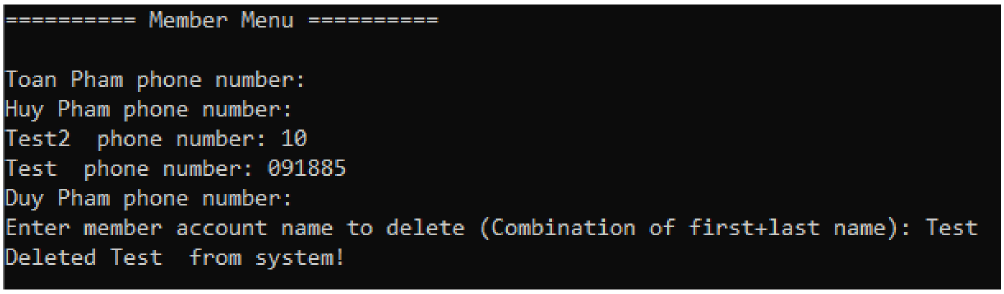


```
==========Staff Menu==========
1. Add a new tool.
2. Add new pieces of an existing Tool.
3. Remove some pieces of a Tool.
4. Register a new member.
5. Remove a member.
6. Find the contact number of a member.
0. Back to Main Menu.
=============================
Please make selection (1-6 or 0 to turn back to menu)
4
Regist Account First Name (cannot be empty):
Regist Account First Name (cannot be empty):
Regist Account First Name (cannot be empty): Account
Regist Account Last Name (cannot be empty):
Regist Account Last Name (cannot be empty):
Regist Account Last Name (cannot be empty): Test
Regist Account Phone Number:
```

5) Remove a member:

a) Successfully remove a member (The user must be added before to remove)

Outcome:



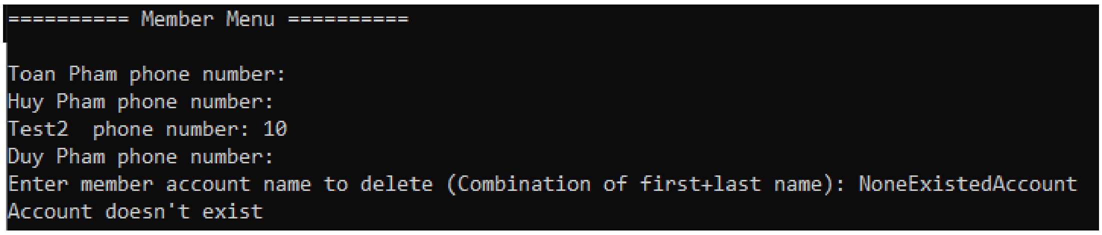
```
========== Member Menu ==========

Toan Pham phone number:
Huy Pham phone number:
Test2  phone number: 10
Test  phone number: 091885
Duy Pham phone number:
Enter member account name to delete (Combination of first+last name): Test
Deleted Test  from system!
```

b) Fail to remove a member:

Outcomes:

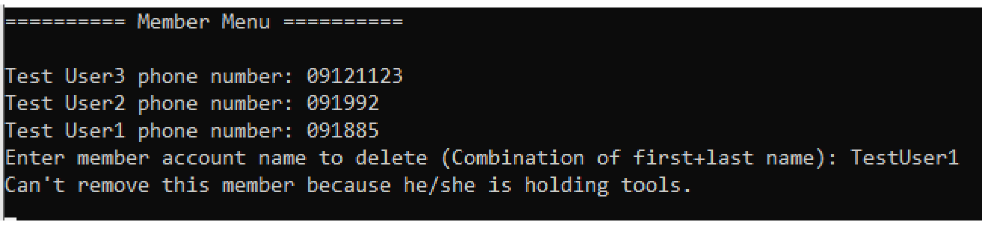
Case: account is not existed.



```
========== Member Menu ==========

Toan Pham phone number:
Huy Pham phone number:
Test2  phone number: 10
Duy Pham phone number:
Enter member account name to delete (Combination of first+last name): NoneExistedAccount
Account doesn't exist
```

Case: the user is currently holding tool(s)



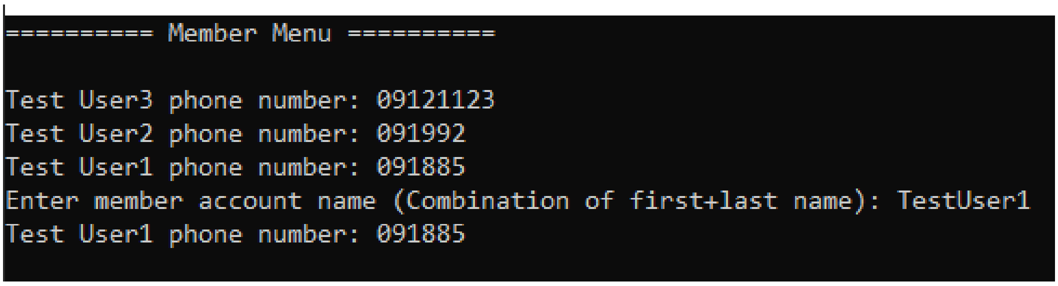
```
========== Member Menu ==========

Test User3 phone number: 09121123
Test User2 phone number: 091992
Test User1 phone number: 091885
Enter member account name to delete (Combination of first+last name): TestUser1
Can't remove this member because he/she is holding tools.
```

6) Find the contact number of a member:

a) Successfully found a member’s phone number: (user is existed and phone number is provided-unless return none)

Outcome:



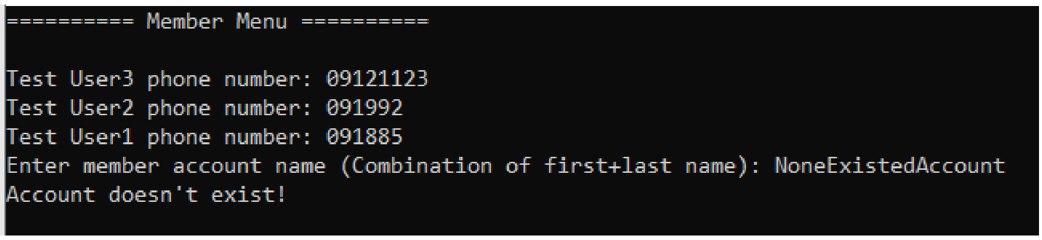
```
========== Member Menu ==========

Test User3 phone number: 09121123
Test User2 phone number: 091992
Test User1 phone number: 091885
Enter member account name (Combination of first+last name): TestUser1
Test User1 phone number: 091885
```

b) Fail to find user’s phone number.

Outcome:

Case: Account name is not existed in the library system



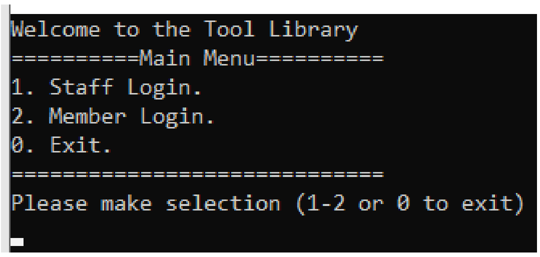
```
========== Member Menu ==========

Test User3 phone number: 09121123
Test User2 phone number: 091992
Test User1 phone number: 091885
Enter member account name (Combination of first+last name): NoneExistedAccount
Account doesn't exist!
```

7) Back to main menu:

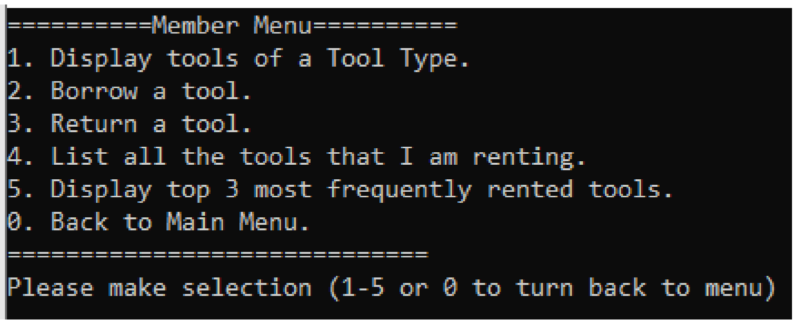
Press “0” to turn back to main menu. (while in staff menu)

Outcome:



```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
```

Member menu functions



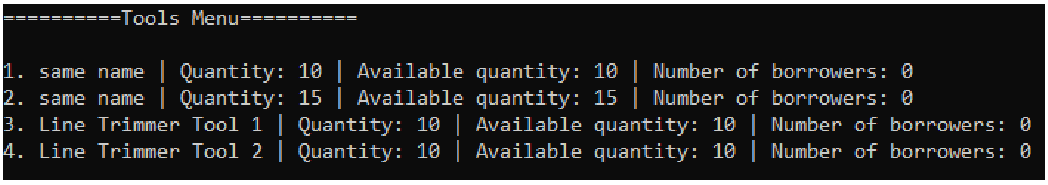
```
==========Member Menu==========
1. Display tools of a Tool Type.
2. Borrow a tool.
3. Return a tool.
4. List all the tools that I am renting.
5. Display top 3 most frequently rented tools.
0. Back to Main Menu.
=============================
Please make selection (1-5 or 0 to turn back to menu)
```

1) Display tools of a tool type

a) Choose any tool type (in this screenshot I choose tool type: “Line trimmers”)

Outcome:

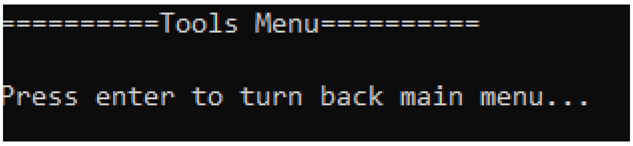
Case: there are tools inside the tool type. (the tools in the screenshot are added into this tool type before)



```
==========Tools Menu==========

1. same name | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
2. same name | Quantity: 15 | Available quantity: 15 | Number of borrowers: 0
3. Line Trimmer Tool 1 | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
4. Line Trimmer Tool 2 | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
```

Case: there is no tools inside the tool type.



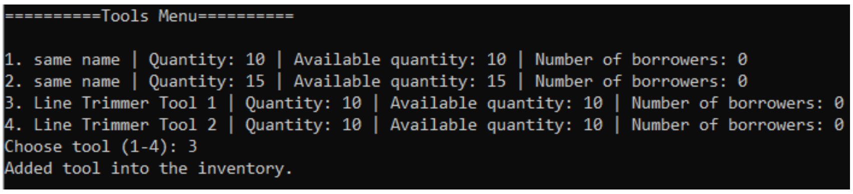
```
==========Tools Menu==========

Press enter to turn back main menu...
```

2) Borrow a tool.

a) Successfully borrowed a tool. (The tools must be first added to system to borrow)

Outcome:



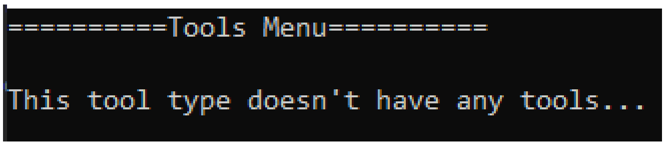
```
==========Tools Menu==========

1. same name | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
2. same name | Quantity: 15 | Available quantity: 15 | Number of borrowers: 0
3. Line Trimmer Tool 1 | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
4. Line Trimmer Tool 2 | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-4): 3
Added tool into the inventory.
```

b) Fail to borrow a tool.

Outcomes:

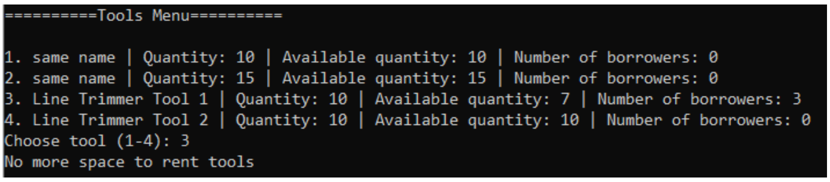
Case: there is no tools to borrow.



```
==========Tools Menu==========

This tool type doesn't have any tools...
```

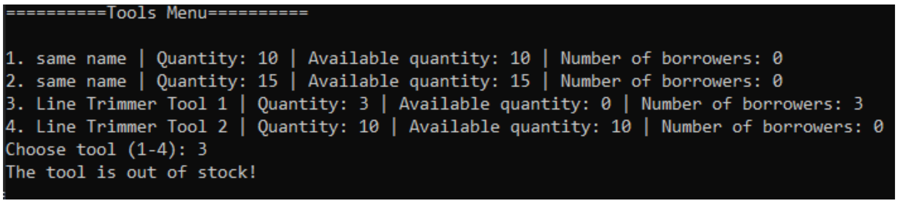
Case: The user already got 3 tools inside the inventory.



```
==========Tools Menu==========

1. same name | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
2. same name | Quantity: 15 | Available quantity: 15 | Number of borrowers: 0
3. Line Trimmer Tool 1 | Quantity: 10 | Available quantity: 7 | Number of borrowers: 3
4. Line Trimmer Tool 2 | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-4): 3
No more space to rent tools
```

Case: The tool's available quantity is 0.



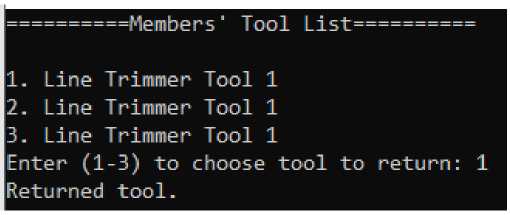
```
==========Tools Menu==========

1. same name | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
2. same name | Quantity: 15 | Available quantity: 15 | Number of borrowers: 0
3. Line Trimmer Tool 1 | Quantity: 3 | Available quantity: 0 | Number of borrowers: 3
4. Line Trimmer Tool 2 | Quantity: 10 | Available quantity: 10 | Number of borrowers: 0
Choose tool (1-4): 3
The tool is out of stock!
```

3) Return a tool.

a) Successfully return a tool (The user must have tools in inventory to return)

Outcome:



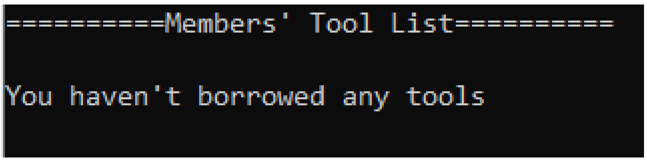
```
==========Members' Tool List==========

1. Line Trimmer Tool 1
2. Line Trimmer Tool 1
3. Line Trimmer Tool 1
Enter (1-3) to choose tool to return: 1
Returned tool.
```

b) Fail to return a tool.

Outcome:

Case: user have not borrowed any tools.



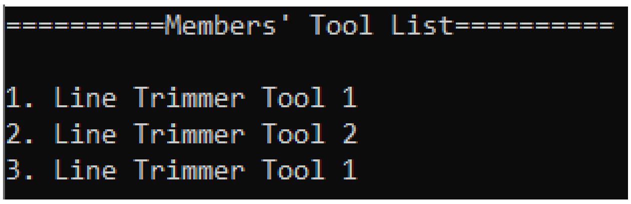
```
==========Members' Tool List==========

You haven't borrowed any tools
```

4) List all the tools being rented.

a) Success to show.

Outcome:

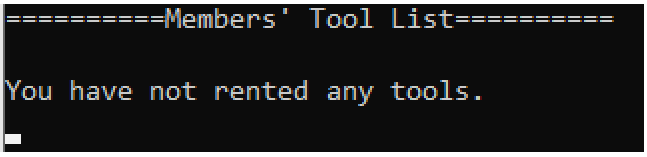


```
==========Members' Tool List==========

1. Line Trimmer Tool 1
2. Line Trimmer Tool 2
3. Line Trimmer Tool 1
```

b) Fail to show. (when there is no tools)

Outcome:



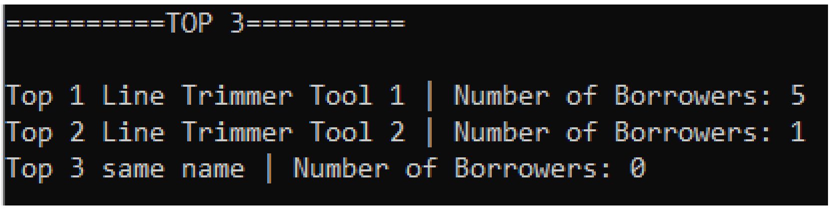
```
==========Members' Tool List==========

You have not rented any tools.
```

5) Display top 3 most frequently rented tools.

a) Success to show (When there are at least 3 tools in the system)

Outcome:



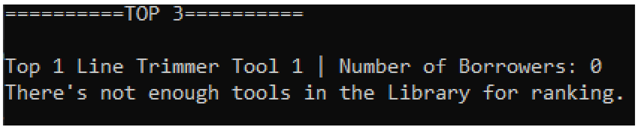
```
==========TOP 3==========

Top 1 Line Trimmer Tool 1 | Number of Borrowers: 5
Top 2 Line Trimmer Tool 2 | Number of Borrowers: 1
Top 3 same name | Number of Borrowers: 0
```

b) Fail to show.

Outcomes:

Case: not enough tools in the library system. (Still show ranking but not fully top 3 tools – for example the screenshot below demonstrates the case where there's only one tool in the system.).

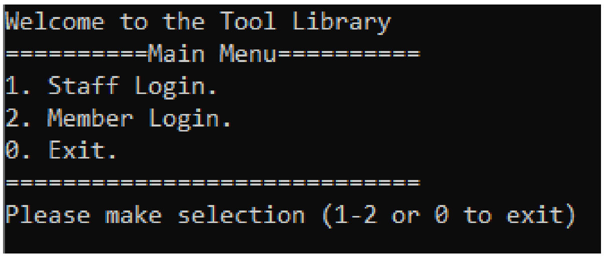


```
==========TOP 3==========

Top 1 Line Trimmer Tool 1 | Number of Borrowers: 0
There's not enough tools in the Library for ranking.
```

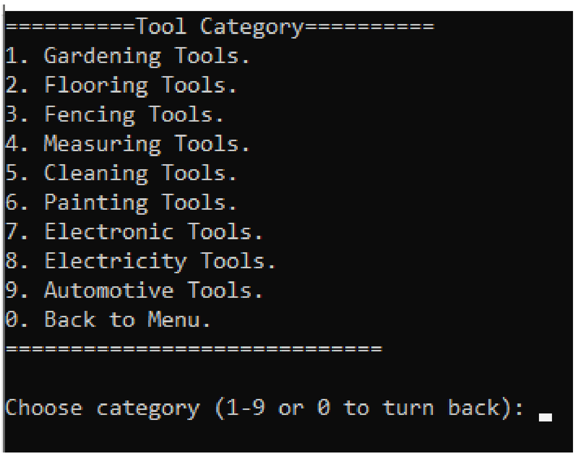
6) Back to main menu (while in member menu)

Outcome:



```
Welcome to the Tool Library
==========Main Menu==========
1. Staff Login.
2. Member Login.
0. Exit.
=============================
Please make selection (1-2 or 0 to exit)
```

Tool categories menu.

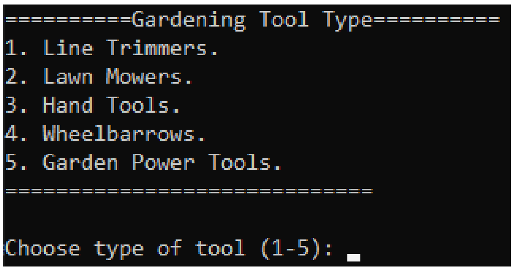


```
==========Tool Category==========
1. Gardening Tools.
2. Flooring Tools.
3. Fencing Tools.
4. Measuring Tools.
5. Cleaning Tools.
6. Painting Tools.
7. Electronic Tools.
8. Electricity Tools.
9. Automotive Tools.
0. Back to Menu.
=============================

Choose category (1-9 or 0 to turn back):
```

1) Choose gardening tools.

Outcome:

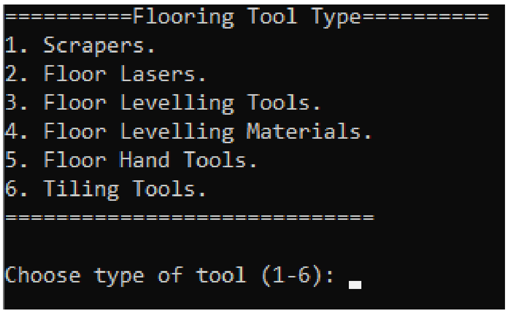


```
==========Gardening Tool Type==========
1. Line Trimmers.
2. Lawn Mowers.
3. Hand Tools.
4. Wheelbarrows.
5. Garden Power Tools.
=============================

Choose type of tool (1-5):
```

2) Choose flooring tools.

Outcome:

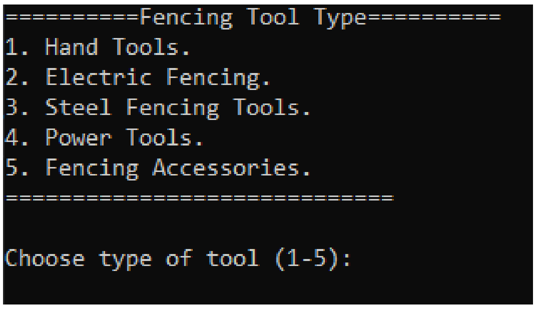


```
==========Flooring Tool Type==========
1. Scrapers.
2. Floor Lasers.
3. Floor Levelling Tools.
4. Floor Levelling Materials.
5. Floor Hand Tools.
6. Tiling Tools.
=============================

Choose type of tool (1-6):
```

3) Choose fencing tools.

Outcome:

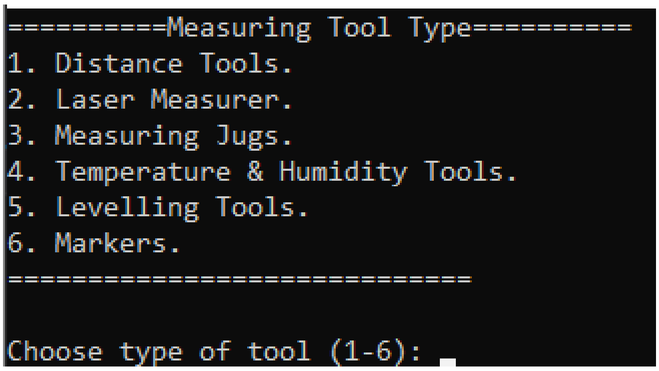


```
==========Fencing Tool Type==========
1. Hand Tools.
2. Electric Fencing.
3. Steel Fencing Tools.
4. Power Tools.
5. Fencing Accessories.
=============================

Choose type of tool (1-5):
```

4) Choose Measuring tools.

Outcome:

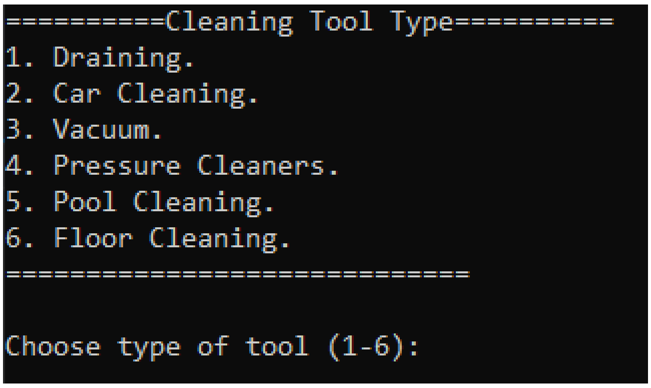


```
==========Measuring Tool Type==========
1. Distance Tools.
2. Laser Measurer.
3. Measuring Jugs.
4. Temperature & Humidity Tools.
5. Levelling Tools.
6. Markers.
=============================

Choose type of tool (1-6):
```

5) Choose cleaning tools.

Outcome:

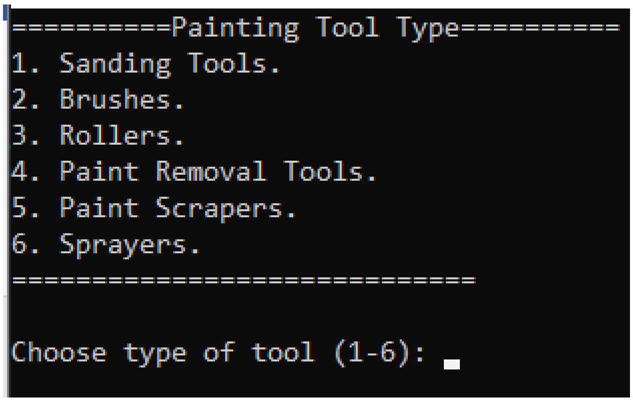


```
==========Cleaning Tool Type==========
1. Draining.
2. Car Cleaning.
3. Vacuum.
4. Pressure Cleaners.
5. Pool Cleaning.
6. Floor Cleaning.
==============================

Choose type of tool (1-6):
```

6) Choose painting tools.

Outcome:

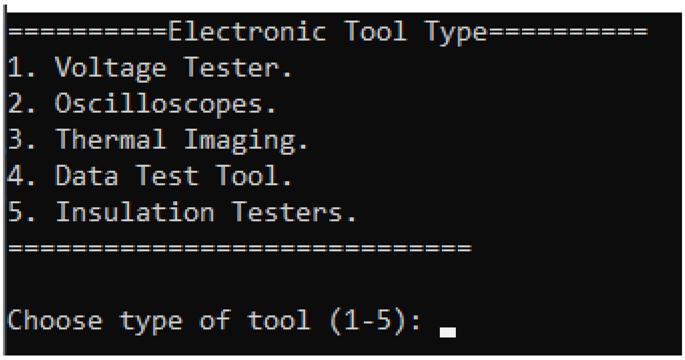


```
==========Painting Tool Type==========
1. Sanding Tools.
2. Brushes.
3. Rollers.
4. Paint Removal Tools.
5. Paint Scrapers.
6. Sprayers.
==============================

Choose type of tool (1-6):
```

7) Choose electronic tools.

Outcome:

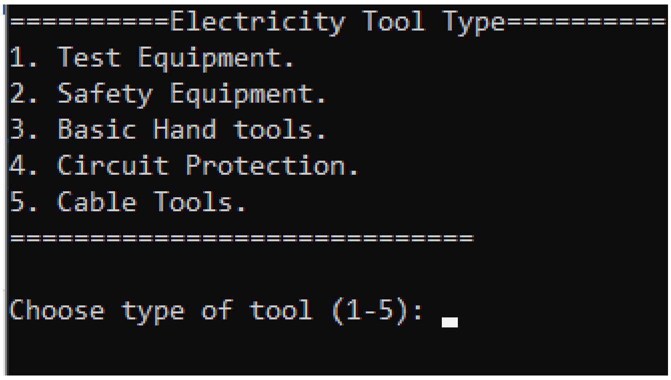


```
==========Electronic Tool Type==========
1. Voltage Tester.
2. Oscilloscopes.
3. Thermal Imaging.
4. Data Test Tool.
5. Insulation Testers.
==============================

Choose type of tool (1-5):
```

8) Choose electricity tools.

Outcome:

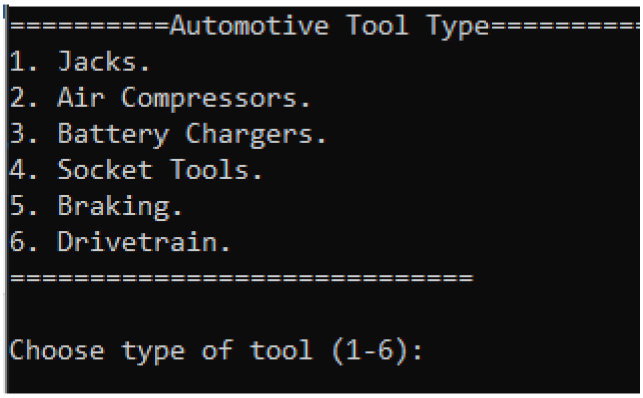


```
==========Electricity Tool Type==========
1. Test Equipment.
2. Safety Equipment.
3. Basic Hand tools.
4. Circuit Protection.
5. Cable Tools.
==============================

Choose type of tool (1-5):
```

9) Choose automotive tools.

Outcome:



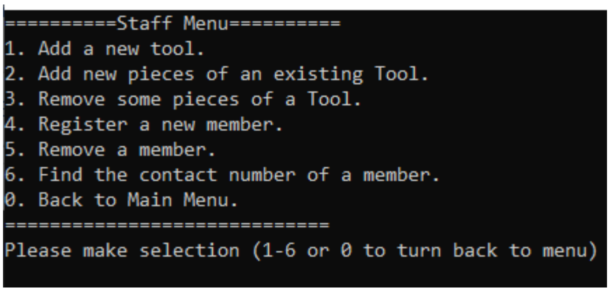
```
==========Automotive Tool Type==========
1. Jacks.
2. Air Compressors.
3. Battery Chargers.
4. Socket Tools.
5. Braking.
6. Drivetrain.
==============================

Choose type of tool (1-6):
```

10) Turn back to menu.

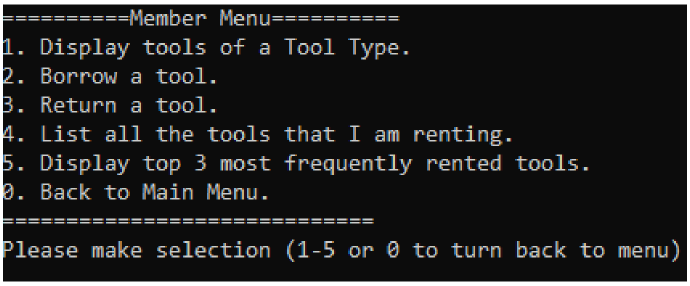
Outcomes:

Case: if the user is the staff



```
==========Staff Menu==========
1. Add a new tool.
2. Add new pieces of an existing Tool.
3. Remove some pieces of a Tool.
4. Register a new member.
5. Remove a member.
6. Find the contact number of a member.
0. Back to Main Menu.
=============================
Please make selection (1-6 or 0 to turn back to menu)
```

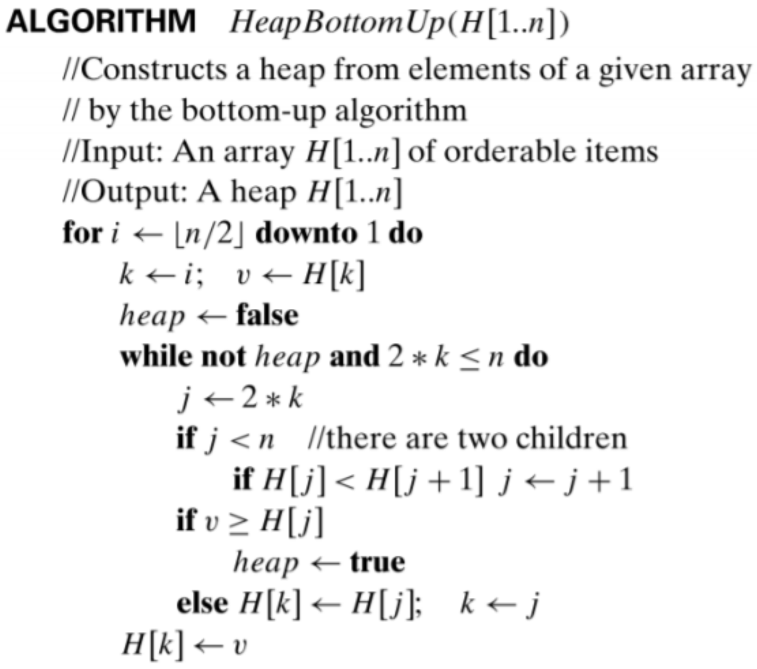
Case: if the user is the member



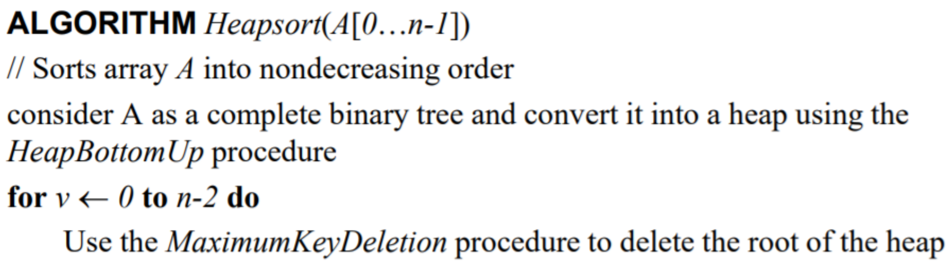
```
==========Member Menu==========
1. Display tools of a Tool Type.
2. Borrow a tool.
3. Return a tool.
4. List all the tools that I am renting.
5. Display top 3 most frequently rented tools.
0. Back to Main Menu.
=============================
Please make selection (1-5 or 0 to turn back to menu)
```

Algorithm

Algorithm design



```
ALGORITHM HeapBottomUp(H[1..n])
    //Constructs a heap from elements of a given array
    // by the bottom-up algorithm
    //Input: An array H[1..n] of orderable items
    //Output: A heap H[1..n]
    for $i \leftarrow \lfloor n/2 \rfloor$ downto 1 do
        $k \leftarrow i$;   $v \leftarrow H[k]$
        $heap \leftarrow$ false
        while not $heap$ and $2 * k \leq n$ do
            $j \leftarrow 2 * k$
            if $j < n$   //there are two children
                if $H[j] < H[j+1]$  $j \leftarrow j+1$
            if $v \geq H[j]$
                $heap \leftarrow$ true
            else $H[k] \leftarrow H[j]$;   $k \leftarrow j$
        $H[k] \leftarrow v$
```



```
ALGORITHM Heapsort(A[0…n-1])
// Sorts array A into nondecreasing order
consider A as a complete binary tree and convert it into a heap using the
HeapBottomUp procedure
for $v \leftarrow 0$ to $n-2$ do
    Use the MaximumKeyDeletion procedure to delete the root of the heap
```

ALGORITHM MaxKeyDelete(A[0…n-1], size)

//This method delete the maximum key and rebuild the whole heap

//Input: An array of orderable items A[0..n-1] and size of that array (a integer value)

//1. Exchange the root's key with the last key K of the heap;

$TempArray \leftarrow A[0]$

$A[0] \leftarrow A[n-1]$

```
$A[n-1] \leftarrow TempArray$
//2. Decrease the heap's size by 1;
$n \leftarrow size - 1$
//3. Heapify the complete binary tree
$heap \leftarrow false$
$k \leftarrow 0$
$v \leftarrow A[0]$
While not heap and $(2 * \mathrm{k} + 1) \leq (\mathrm{n} - 1)$ do
    $j \leftarrow 2 * \mathrm{k} + 1$                //the left child of k
    If $j < \mathrm{n} - 1$ do                  //k has 2 children
        If $A[j] < A[j+1]$ do
            $j \leftarrow j + 1$        //j is the larger child of k
        If $v \geq A[j]$ do
            $heap \leftarrow true$
        else do
            $A[k] \leftarrow A[j]$
            $k \leftarrow j$
$A[k] \leftarrow v$
```

```
ALGORITHM displayTopTHree()
//This method rearrange a toolcollection object inside the tool library system then choose out 3 most frequently rented tool. Or in another hand, this method rearranges a list of orderable items then return 3 items with highest values.
//step 1. Create an empty array to store all existed values inside the system.
$v \leftarrow A[\emptyset]$   //empty array
//step 2. Scan through the system to spot out existed tool and add to the created array. By using this block of code the array will have value v[1…n]
For each ToolCategory [1…n] in Categories [1…9] do
    For each ToolType [1…n] in ToolCategory [1…n] do
        For each Tool in ToolType [1…n] do
            $v.add(Tool)$            //add found tools into the empty array
```

//**step 3**. Sort the array in ascending order using HeapSort function

```
Heapsort($v$)
```

//**step 4**. Return 3 items with highest values. The array will now be v[1…n] with ascending values.

```
$top1 \leftarrow v[n-1]$                // “n” is the number of items inside the array
$top2 \leftarrow v[n-2]$
$top3 \leftarrow v[n-3]$
```

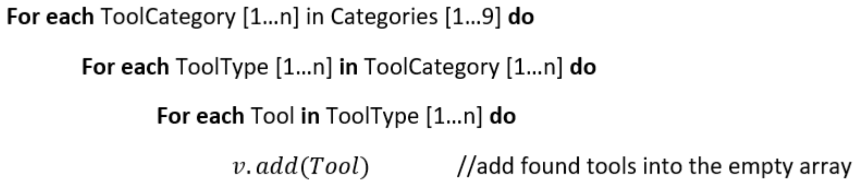
Algorithm analysis

STEP 1: (C1)

Take: 1

STEP 2: (C2)

- According to the pseudocode provided above. I can identify the **basic operation** of the function is $v.add(Tool)$.



```
For each ToolCategory [1…n] in Categories [1…9] do
    For each ToolType [1…n] in ToolCategory [1…n] do
        For each Tool in ToolType [1…n] do
            $v.add(Tool)$          //add found tools into the empty array
```

- This basic operation is nested in 3 for-each loops (loop of each category in categories, tool type in each category and tool in each tool type).

  - From the most inner-loop the worst case is when there’s n tools available in each tool type (No tool type is empty). Therefore, the basic operation will happen “n” times. => **n times worst case**

  - Middle-loop (“**For each** ToolType [1…n] **in** ToolCategory [1…n] **do**”), based on the category and tool type structure, each category will have between minimum 5 and maximum 6 tools. Therefore, the worst-case for this loop will be a category which has 6 tool types. => **6 times worst case**

  - Outer-most-loop has the fix value of 9 categories. => **9 times run**

Total: n x 6 x 9 = 54n => step 2 will take 54n

STEP 3: (C3)

using Heapsort() method will have **O(n Log n)**

STEP4: (C4)

Take: 3

The worst-case time complexity in big-O notation of DisplayTopTHree() method:

$T(n) = C1 + C2 + C3 + C4 = 1 + 54n + O(n \log n) + 3 = 4 + 54n + O(n \log n)$

- ⇨ Worst case ∈ **$O(n \log n)$**