Assignment Case	
OOP using JavaH6	
Periode Berlaku Semester Ganjil 2023/2024 ***Valid on** Odd Year 2023/2024*	**Software Laboratory Center** **Assistant Recruitment 24-1**

Note: Please focus on the main logic and main feature!
(Splash screen and design are not scored)

Soal

Case

SeRene Hairsalon

Serene Hair Salon is a calm and welcoming place where people can take a break and enjoy great hair services. In our peaceful setting, friendly hairdressers and a helpful assistant and cashier are ready to make its customer's visit smooth and enjoyable. However, in daily basis the hairsalon is full of customers thus making it hard to manage and ensuring customers have a great and smooth experience. You as a profesional programmer are ask to create a simple program that keeps track of what's happening in the salon. The application has the following requirements:

- **Starting Page** (**Menu 1**)
  - This menu contains of **3** menus, which are **Play New Hairsalon**, **Hair Score**, and **Exit**.
  - **Prompt** user to **input chosen menu**. **Validate** the input must **between 1 and 3 inclusively**.

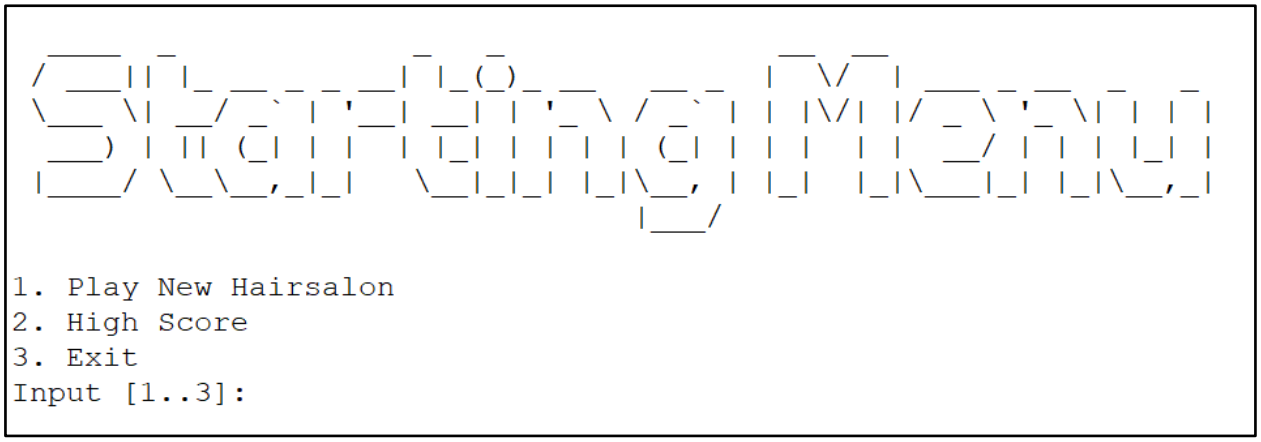

Figure 1. Starting Page

1. If user choose **Play New Hairsalon** (**Menu 1**), then:
   - **Prompt** user to **input name** for the salon. **Validate** the inputted name must be **5 to 20 characters**. If the input is not as expected, show an error message.

- When the user has inputted the name of the hair salon, the program must be able to **create a hair salon** with starting status **5 seats, 2 cashiers, 2 hairdressers, 2 assistants, 1200 worth of money, and 0 for score** and **redirect** user to **game menu**. Make sure that **there's only one hair salon that is currently playing**.

```
Input your hairsalon's name [5..20 characters]:
se
name must be 5 to 20 characters

Input your hairsalon's name [5..20 characters]:
serene
```

Figure 2. Input name

2. If user choose **High Score** (**Menu 2**), then:
   - **Show Top 3 Hairsalon High score** with these following requirements**:**
     - Must be in **ascending order** according to its **score**.
     - Show the name and score of the hair salon.
     - Only show **top 3** hair salon.
     - One hair salon **must not** be shown twice.
   - Prompt the user to input **1** to go back to the starting page.

```
 _   _ _       _
| | | (_) __ _| |__  ___  ___ ___  _ __ ___
| |_| | |/ _` | '_ \/ __|/ __/ _ \| '__/ _ \
|  _  | | (_| | | | \__ \ (_| (_) | | |  __/
|_| |_|_|\__, |_| |_|___/\___\___/|_|  \___|
         |___/

Name : Obewraw dadaw, Score: 350
Name : serene, Score: 325
Name : jopjop salon, Score: 175
Enter 1 to back...
```

Figure 3. Highscore Page

3. If user choose **Exit** (**Menu 3**), then:
    - Show splash logo.
    - **Terminate** console.

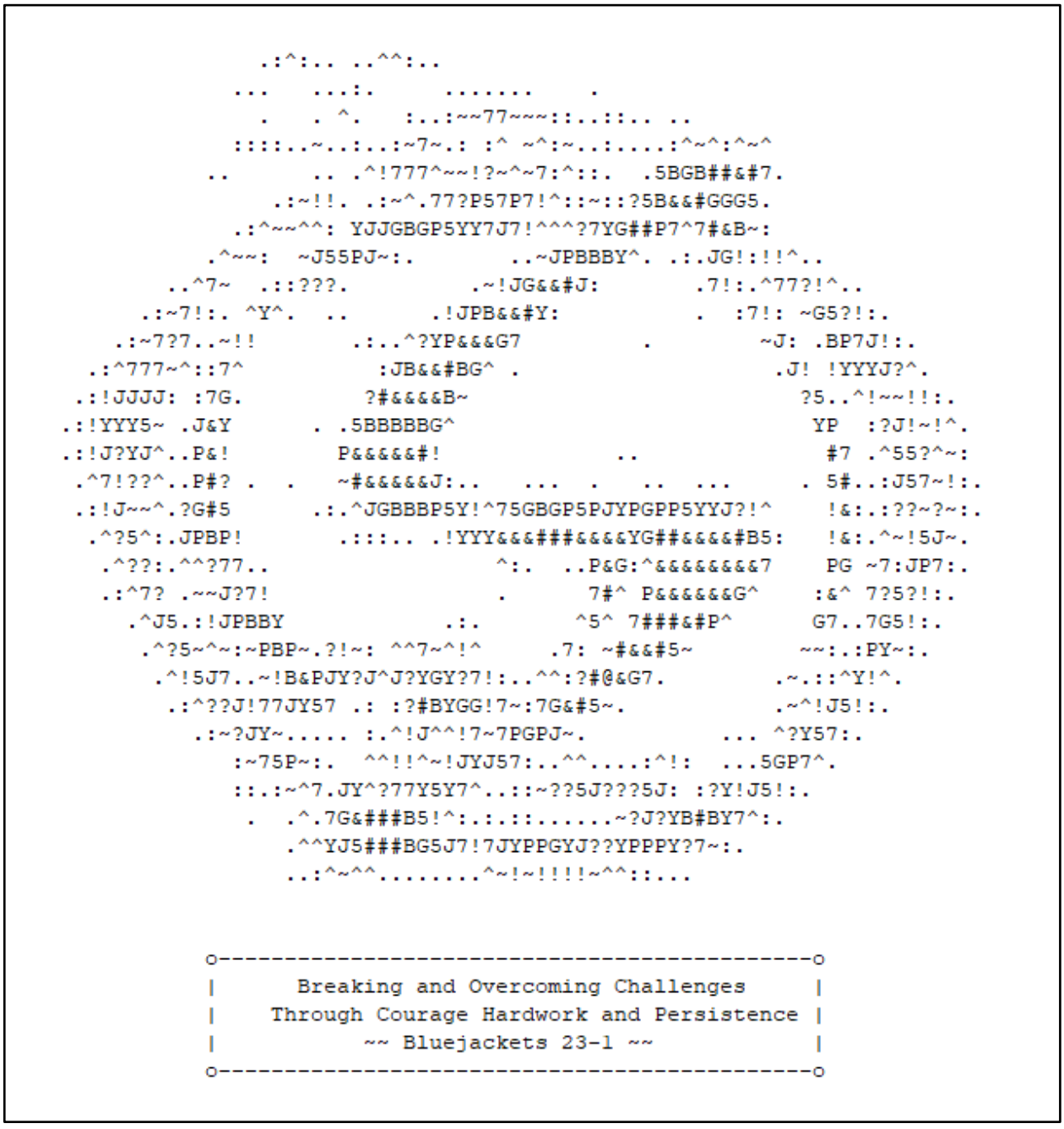

Figure 4. Exit Page

- **Game Page** (**Menu 2**)
  - This page is **shown right after** the player has **finished inputting** the name of the hair salon.
  - This page **must show**:
    - Hair salon status (hair salon's name, hair salon's money, hair salon's score, hair salon's size/seats).
    - All of hair salon's activity.
  - This page must be **refreshed every one second**.
  - If a user input an enter. Users will be directed to pause menu and all activity must be on hold.

```
Hairsalon 'serene' is open!

Status
Money :1200
Score :0
Size :5
==================================================================================================
| Customer                         | Assistant              | Hairdresser            | Cashier        |
==================================================================================================
| JG (21) hair being washed <YY>   | YY washing hair <JG>   | OD idle                | OI idle        |
| RH (22) hair being washed <ED>   | ED washing hair <RH>   | JR idle                | SG idle        |
==================================================================================================
```

Figure 5. Game Menu

- **Hair salon**
  - As mentioned earlier, new hair salons have a starting status of **5 seats, 2 cashiers, 2 hairdressers, 2 assistants, 1200 worth of money, and 0 for score.**
  - When a seat/slot is available, there's a **25% chance** a new customer will come to fill that slot.
- **Customer**
  - A customer has **2** attributes, an **initial and tolerance rate**.
    - The initial must consist of **2** characters in uppercase. The initial is generated **randomly** and must be **unique** (only one person has the initial in the entire hair salon)
    - Every customer has **22** worth of **tolerance rate**.
  - Customer **6** status.
    - Waiting assistant: when a customer has arrived at the hair salon, an assistant should assist them by washing their hair. But if all **assistants are occupied**, the

customer status is '**waiting assistant'** and the customer must be in a **waiting line**. While waiting for an assistant, **every 1 seconds** the customer's tolerance rate is **reduced by one**.

- Hair washed: when an assistant is finally **available** and ready to serve a customer, the customer's status should change to '**hair washed**' and **display the assistant's initial** that is currently serving the customer. This process should take about **(7 – the assistant's speed) seconds**.
- Waiting hairdresser: after the assistant has finished the process, **both assistant and the customer** must wait for the available hairdresser in a **waiting line**. While waiting, **every 4 seconds** the customer's tolerance rate is **reduced by one** and the status should be '**waiting hairdresser**'.
- Hair cut: when a hairdresser is **available** and ready, the customer's status should change to '**hair cut**' and **display the hairdresser's initial** that is currently serving the customer. This process should take about **(7 – the hairdresser's speed) seconds**.
- Waiting cashier: after the process is completed, **both hairdresser and the customer** should be in the cashier **waiting line**. While waiting, **every 4 seconds** the customer's tolerance rate is **reduced by one** and the customer status should be '**waiting cashier**'.
- Pay: when a cashier is **available** and ready to serve the customer, the customer's status should change to '**pay**' and **display the cashier's initial** that is currently serving the customer. This process should take about **(7 – the cashier's speed) seconds**. After the process is done, **add (skill hairdresser that has served that customer * 15)** to the **hair salon's money and score** and customer **leaves** the hair salon.

o When a customer's tolerance rate **reaches 0**, the customer **leaves** the hair salon **regardless, reduce** hair salon's score by **200,** and **change** the **serving staff's status** to **idle**.

- **Cashier**
  - A cashier has **2** attributes, **an initial and speed**.
    - The initial must consist of **2** characters in uppercase. The initial is generated **randomly** and must be **unique** (only one person has the initial in the entire hair salon)
    - The starting **speed** for a new cashier is **1** (if the hair salon is also new).
  - A cashier has **2** status.
    - Idle: when the cashier has **no waiting line** and **no customer left to serve**, the cashier's status changes to '**idle'**.
    - Pay: when the cashier is **serving a customer**, the cashier's status changes to '**pay**' and **display the customer's initial**. This process should take about **(7 – the cashier's speed) seconds**.
- **Assistant**
  - An assistant has **2** attributes, an **initial and speed**.
    - The initial must consist of **2** characters in uppercase. The initial is generated **randomly** and must be **unique** (only one person has the initial in the entire hair salon)
    - The starting **speed** for a new assistant is **1** (if the hair salon is also new).
  - An assistant has **3** status.
    - Idle: when the assistant has **no waiting line** and **no customer left to serve**, the assistant's status changes to '**idle**'.
    - Washing hair: when the assistant is **serving a customer**, the assistant's status changes to '**washing hair**' and **display the customer's initial**. This process should take about **(7 – the assistant's speed) seconds**.
    - Waiting hairdresser: after the assistant has **completed** washing the customer's hair, **both assistant and the customer** must wait for **available** hairdresser to take over. In this state, the assistant status should be '**waiting hairdresser**'.

- **Hairdresser**
  - A hairdresser has **3** attributes, **an initial, skill and speed**.
    - The initial must consist of **2** characters in uppercase. The initial is generated **randomly** and must be **unique** (only one person has the initial in the entire hair salon)
    - The starting **speed and skill** for a new hairdresser is **1** (if the hair salon is also new).
  - A hairdresser has **3** status.
    - Idle: when the hairdresser has **no waiting line** and **no customer** left to serve, the hairdresser's status changes to '**idle**'.
    - Cutting hair: when the hairdresser is **serving a customer**, the hairdresser's status changes to '**cutting hair**' and **display the customer's initial**. This process should take about **(7 – the hairdresser's speed) seconds**.
    - Waiting cashier: after the hairdresser has **completed** cutting the customer's hair, **both hairdresser and the customer** must wait for **available** cashier to take over. In this state, the hairdresser status should be '**waiting cashier**'.
- **Pause Menu**
  - When a user inputs an **enter** while the game is **playing**, the game should be **on hold** and **display** the pause menu. The pause menu consists of **3** options, **upgrade hairsalon**, **close hairsalon**, and **back.**

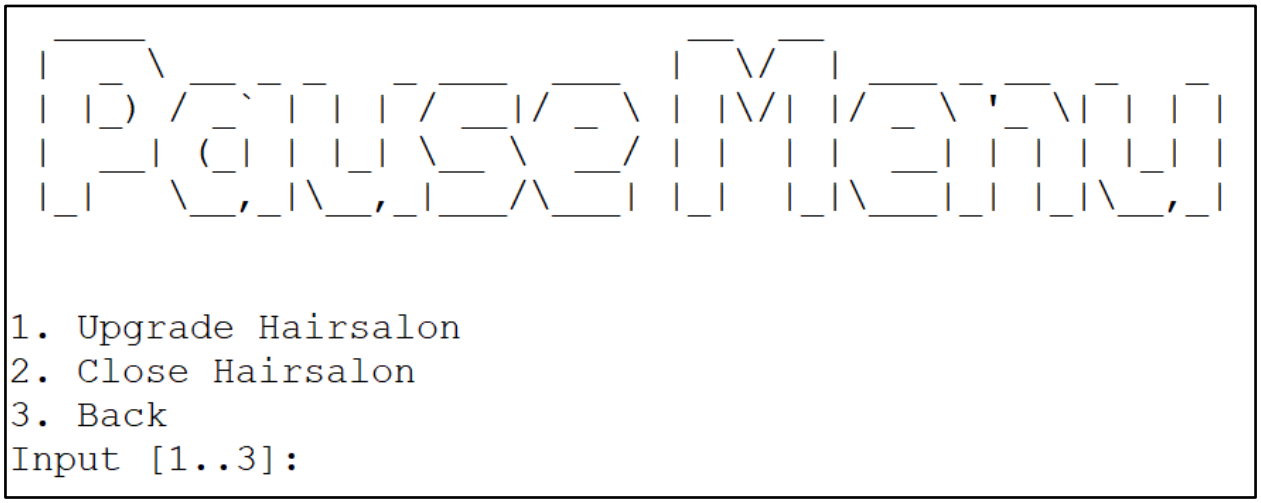

Figure 6. Pause Menu

1. If user choose **Upgrade Hairsalon** (**Menu 1**), then:
    - **Redirect** user to upgrade menu.
2. If user choose **Close Hairsalon** (**Menu 2**), then:
    - **Redirect** user to starting menu, **add the hair salon's name and hair salon's score** to 'highscore.txt', and make sure that there's **no hair salon** that is **currently active**.
3. If user choose **Back** (**Menu 3**), then:
    - **Redirect** user to game menu and **resume** the game.

➢ **Upgrade Menu**

- The pause menu consists of **9** options, **increase hairsalon's seat**, **upgrade assistant's speed, upgrade hairdresser's speed, upgrade cashier's speed, upgrade hairdresser's skill, hire new assistant, hire new cashier, hire new hairdresser,** and **back.**

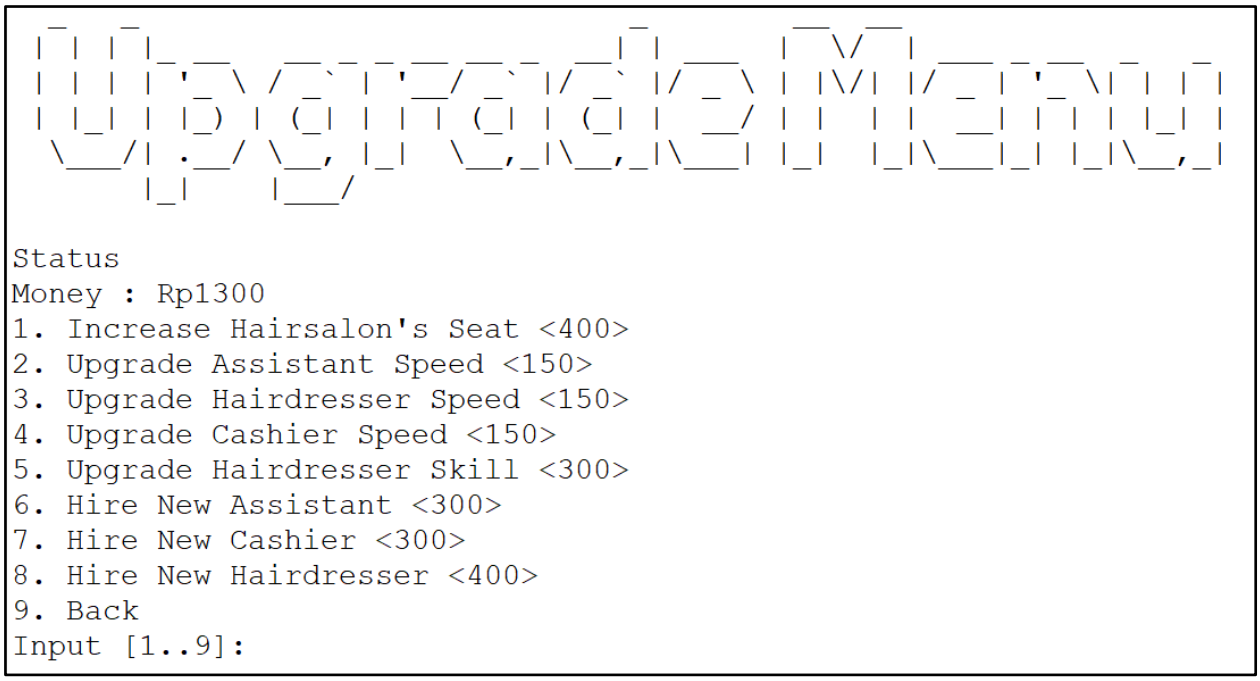

Figure 7. Upgrade Menu

1. If user choose **Upgrade Hairsalon's Seat** (**Menu 1**), then:
    - The upgrade will cost **400**, **validate** that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **increase** hair salon's seats **by one**. If the money doesn't meet the requirements, **show an error message**.

2. If user choose **Upgrade Assistant Speed** (**Menu 2**), then:
    - The upgrade will cost **150**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **increase** all assistants' speed **by one**. If the money doesn't meet the requirements, **show an error message**.
    - Also validate that the **maximum speed** for assistants is **6**. So, if all the requirements above are met but the assistant is **already at maximum speed**, the upgrade **cannot be done** and **show an error message**.
3. If user choose **Upgrade Hairdresser Speed** (**Menu 3**), then:
    - The upgrade will cost **150**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **increase** all hairdressers' speed **by one**. If the money doesn't meet the requirements, **show an error message**.
    - Also validate that the **maximum speed** for hairdressers is **6**. So, if all the requirements above are met but the assistant is **already at maximum speed**, the upgrade **cannot be done** and **show an error message**.
4. If user choose **Upgrade Cashier Speed** (**Menu 4**), then:
    - The upgrade will cost **150**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **increase** all cashiers' speed **by one**. If the money doesn't meet the requirements, **show an error message**.
    - Also validate that the **maximum speed** for cashiers is **6**. So, if all the requirements above are met but the assistant is **already at maximum speed**, the upgrade **cannot be done** and **show an error message**.
5. If user choose **Upgrade Hairdresser Skill** (**Menu 5**), then:
    - The upgrade will cost **300**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **increase** all hairdressers' skill **by one**. If the money doesn't meet the requirements, **show an error message**.

6. If user choose **Hire New Assistant** (**Menu 6**), then:
    - The upgrade will cost **300**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **add a new assistant** to the hair salon, make sure the new assistant **has the same level of speed** as the other assistant. If the money doesn't meet the requirements, **show an error message**.
7. If user choose **Hire New Cashier** (**Menu 7**), then:
    - The upgrade will cost **300**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, **reduce** the money according to the price and **add a new cashier** to the hair salon, make sure the new cashier **has the same level of speed** as the other cashier. If the money doesn't meet the requirements, **show an error message**.
8. If user choose **Hire New Hairdresser** (**Menu 8**), then:
    - The upgrade will cost **400**, validate that the hair salon's money **must be enough** to upgrade. If it's enough, reduce the money according to the price and **add a new hairdresser** to the hair salon, make sure the new hairdresser **has the same level of speed and skill** as the other hairdresser. If the money doesn't meet the requirements, **show an error message**.
9. If user choose **Back** (**Menu 9**), then:
    - **Redirect** user to pause menu.

Please run the EXE file to see the sample program.