



**FACULTY OF ENGINEERING AND TECHNOLOGY  
DEPARTMENT OF COMPUTER ENGINEERING**

**ENCS4370 - COMPUTER ARCHITECTURE**

**First semester 2024/2025**

---

**First Project – Interactive Monthly Calendar Application**

Prepared by:

Tariq Odeh – 1190699

Wasim Atta – 1200920

Sec: 2

Instructor: Dr. Aziz Qaroush

Date: 8<sup>th</sup> December 2023

---

## Abstract

This project aims to develop a MIPS assembly code that facilitates the viewing, editing, and management of appointments within a monthly calendar. The envisioned application is designed to offer a user-friendly interface, enabling users to seamlessly interact with calendar functionalities. Users will have the capability to add, edit, and view appointments for specific dates, enhancing the overall user experience in appointment scheduling and organization.

## Table of Contents

1. Assumptions.....	5
2. To Do list .....	5
3. Teamwork .....	5
4. Calculations.....	6
5. Testing examples .....	7
5.1. The Start .....	7
5.2. View the Calendar .....	8
5.2.1. Per day .....	8
5.2.2. Per set of days.....	9
5.2.3. Given slot in a given day .....	10
5.2.4. Back to main menu .....	11
5.3. View Statistics.....	12
5.4. Add a new appointment.....	13
5.5. Delete an appointment.....	14
5.6. Exit .....	15
6. Appendix.....	16

## List of Figures

Figure 5-1: File input (calendar.txt).....	7
Figure 5-2: The main menu.....	7
Figure 5-3: The view calendar menu .....	8
Figure 5-4: View day 4 .....	8
Figure 5-5: View day 14 (empty day).....	8
Figure 5-6: View set of days (1, 6, and 18) .....	9
Figure 5-7: View set of days (26, 31, and 20) .....	9
Figure 5-8: View set of days .....	9
Figure 5-9: View slot 8-12 in day 26.....	10
Figure 5-10: View slot 8-10 in day 27 (empty day).....	10
Figure 5-11: View slot 8-4 in day 31 .....	10
Figure 5-12: View slot (invalid input) .....	11
Figure 5-13: Back to main menu .....	11
Figure 5-14: View Statistics.....	12
Figure 5-15: View Statistics after edit .....	12
Figure 5-16: Add a new appointment in the middle of day .....	13
Figure 5-17: Add a new appointment in empty day .....	13
Figure 5-18: Add a new appointment in end of the day .....	13
Figure 5-19: Add a new appointment in start of the day .....	13
Figure 5-20: Delete specific slot.....	14
Figure 5-21: Delete from start of slot .....	14
Figure 5-22: Delete from end of slot.....	14
Figure 5-23: Delete from slot (invalid input).....	15
Figure 5-24: Exit from prgram.....	15
Figure 6-1: Design view algorithm -1.....	16
Figure 6-2: Design view algorithm -2.....	16
Figure 6-3: Design delete algorithm .....	17

## List of Tables

Table 3-1: Teamwork.....	5
--------------------------	---

## 1. Assumptions

- We take October 2023 as the assumption for our case.
- The calendar is for a month and this month has 31 days.
- The first day of the month is Sunday and the last day is Tuesday.
- We assume that the calendar.txt file is in the right formula.

## 2. To Do list

- Create an input file.
- Make the menu.
- Read the input file.
- View the calendar.
  - Per day.
  - Per set of days.
  - Given slot in a given day.
- Add a new appointment.
- Delete an appointment.
- View Statistics.
  - Number of lectures (in hours).
  - Number of OH (in hours).
  - Number of Meetings (in hour).
  - Average lectures per day.
  - Ratio between total number of hours reserved for lectures and total number of hours reserved OH.
- Store calendar in text file.

## 3. Teamwork

Table 3-1: Teamwork

Task	Team member
View the calendar	Tariq
Delete an appointment	Tariq
Create an input file / Read file	Tariq
Make the menu	Tariq / Wasim
Store calendar in text file	Tariq / Wasim
Add a new appointment.	Wasim
View Statistics	Wasim

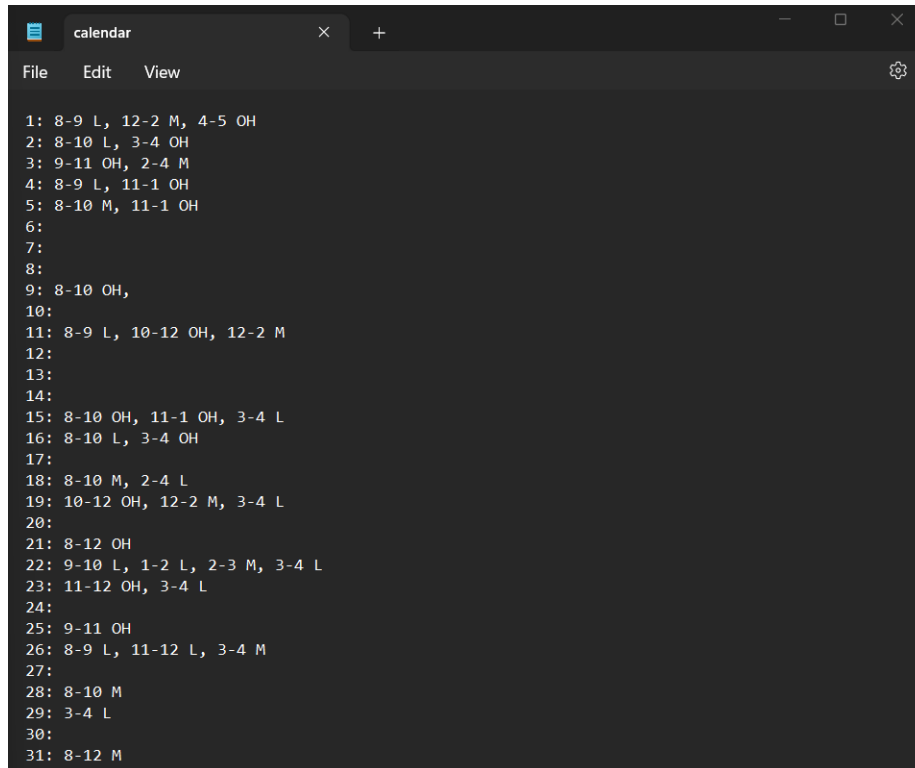
## 4. Calculations

- **calendar\_mem (Memory space to store the calendar)**
  - 31: This represents the number of days in the month, as there are 31 lines.
  - 10: This is an estimated size for each appointment.
  - 9: This represents the maximum number of appointments per day.
  - Total size =  $31 \times 10 \times 9$
  - Total size = 2790
- **input\_string (Maximum length of the input string)**
  - The sequence "1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31" has 91 characters, including digits and commas.
- **day\_buffer (Buffer to store the day as a string)**
  - 10: This is an estimated size for each appointment.
  - 9: This represents the maximum number of appointments per day.
  - Total size =  $10 \times (9+1)$
  - Total size = 100

## 5. Testing examples

### 5.1. The Start

When the program is run, the program reads the Calendar.txt file and displays the main menu for the user to choose the operation he wants.



```
calendar
File Edit View
1: 8-9 L, 12-2 M, 4-5 OH
2: 8-10 L, 3-4 OH
3: 9-11 OH, 2-4 M
4: 8-9 L, 11-1 OH
5: 8-10 M, 11-1 OH
6:
7:
8:
9: 8-10 OH,
10:
11: 8-9 L, 10-12 OH, 12-2 M
12:
13:
14:
15: 8-10 OH, 11-1 OH, 3-4 L
16: 8-10 L, 3-4 OH
17:
18: 8-10 M, 2-4 L
19: 10-12 OH, 12-2 M, 3-4 L
20:
21: 8-12 OH
22: 9-10 L, 1-2 L, 2-3 M, 3-4 L
23: 11-12 OH, 3-4 L
24:
25: 9-11 OH
26: 8-9 L, 11-12 L, 3-4 M
27:
28: 8-10 M
29: 3-4 L
30:
31: 8-12 M
```

Figure 5-1: File input (calendar.txt)

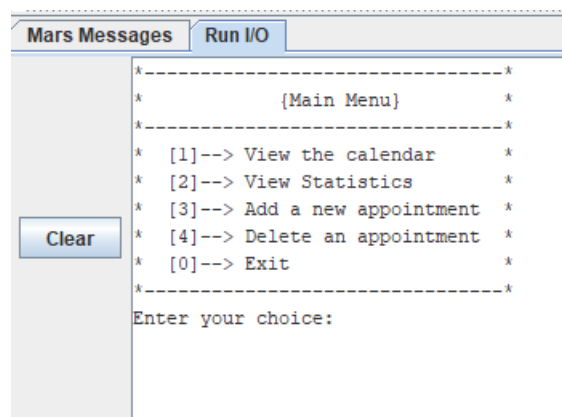


Figure 5-2: The main menu



## 5.2. View the Calendar

When the user chooses the view calendar operation, the program displays the view calendar menu.

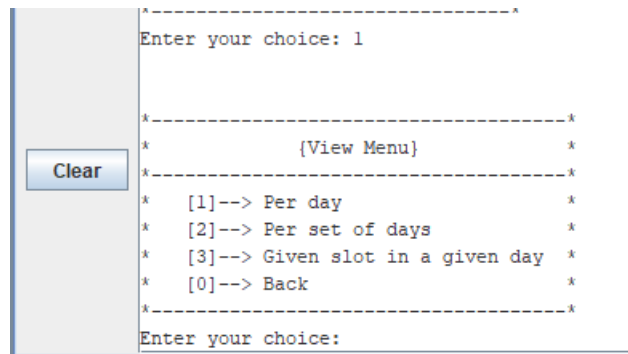


Figure 5-3: The view calendar menu

### 5.2.1. Per day

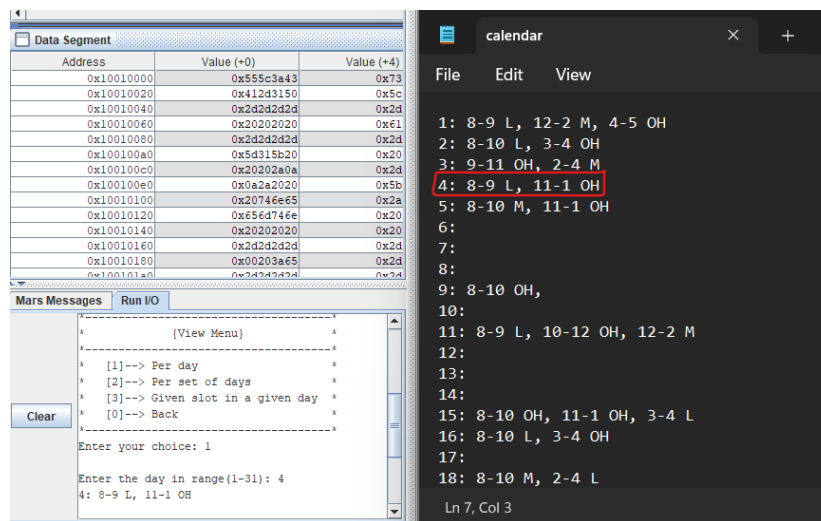


Figure 5-4: View day 4

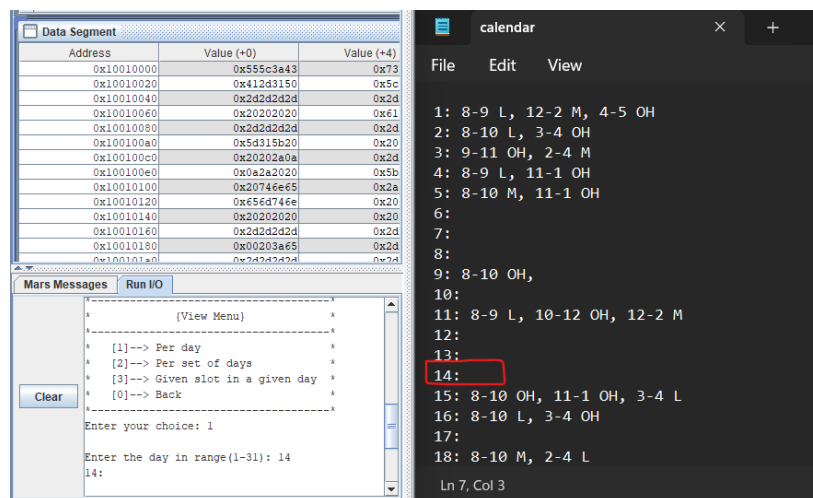


Figure 5-5: View day 14 (empty day)

### 5.2.2. Per set of days

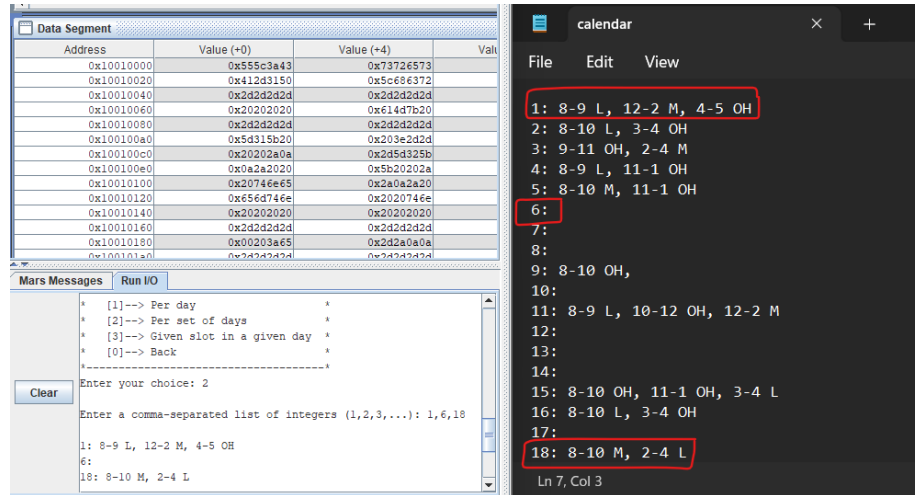


Figure 5-6: View set of days (1, 6, and 18)

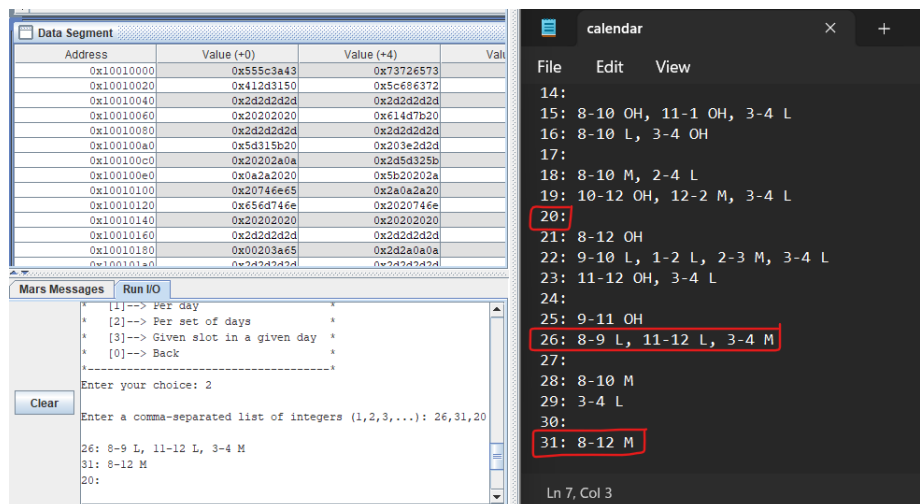


Figure 5-7: View set of days (26, 31, and 20)

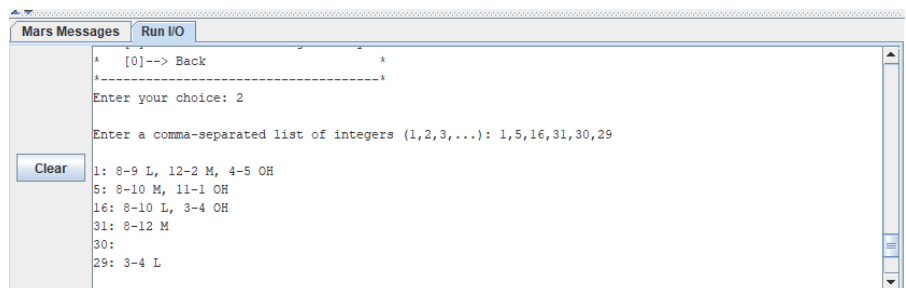


Figure 5-8: View set of days

### 5.2.3. Given slot in a given day

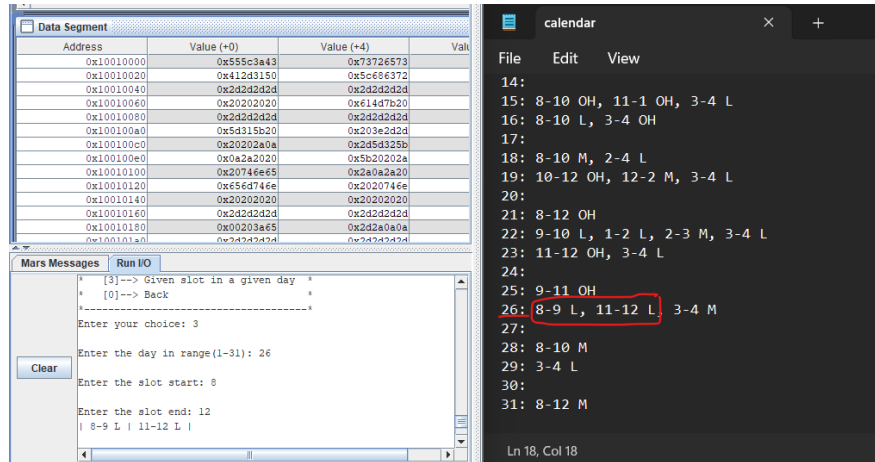


Figure 5-9: View slot 8-12 in day 26

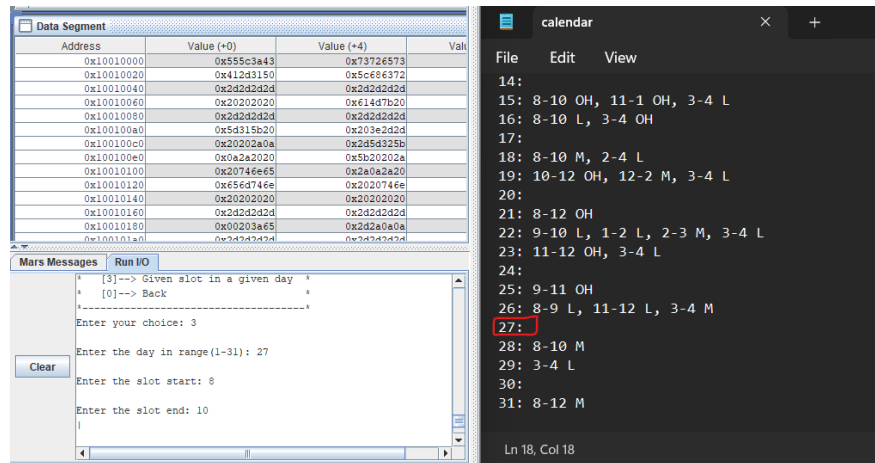


Figure 5-10: View slot 8-10 in day 27 (empty day)

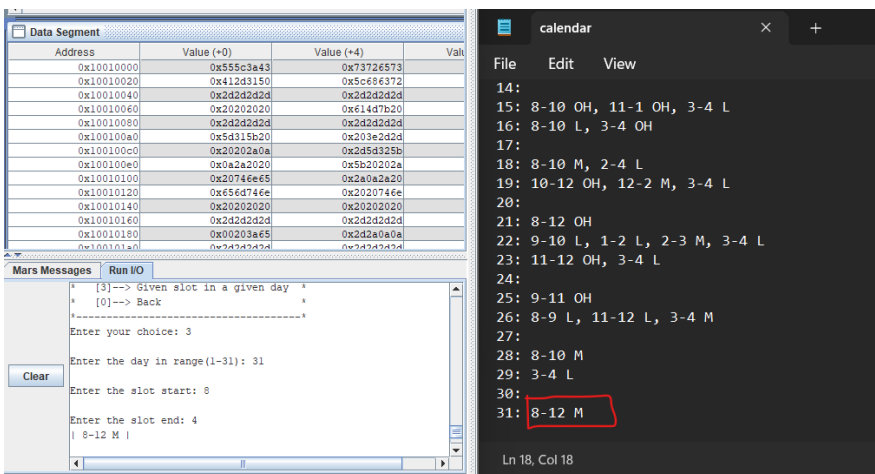


Figure 5-11: View slot 8-4 in day 31

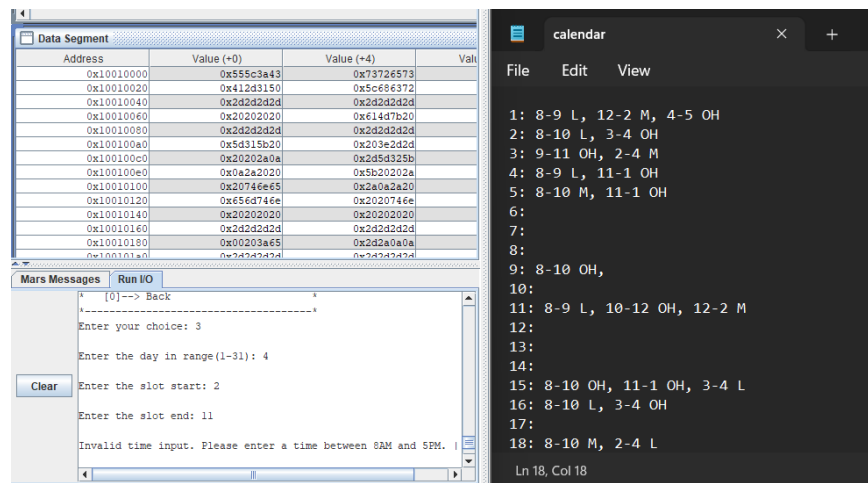


Figure 5-12: View slot (invalid input)

## 5.2.4. Back to main menu

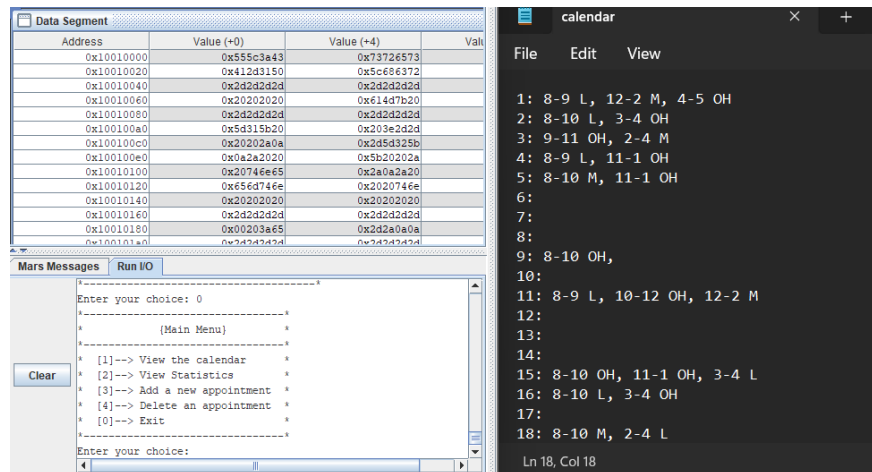


Figure 5-13: Back to main menu

### 5.3. View Statistics

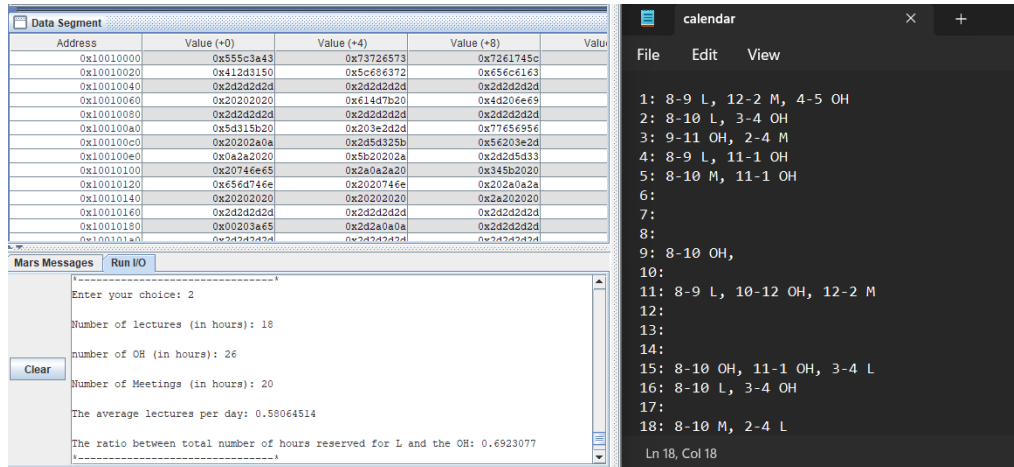


Figure 5-14: View Statistics

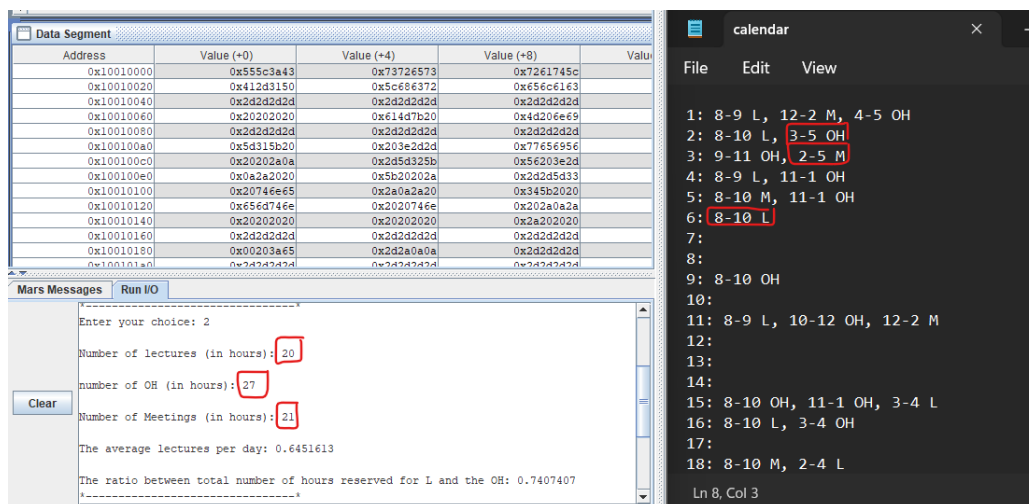
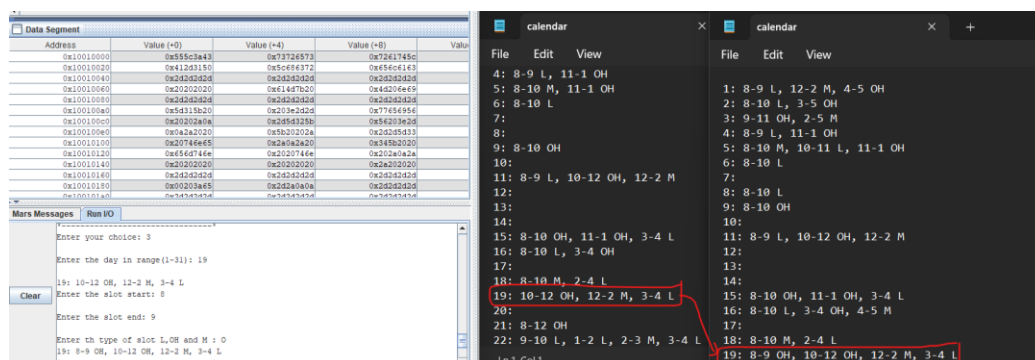
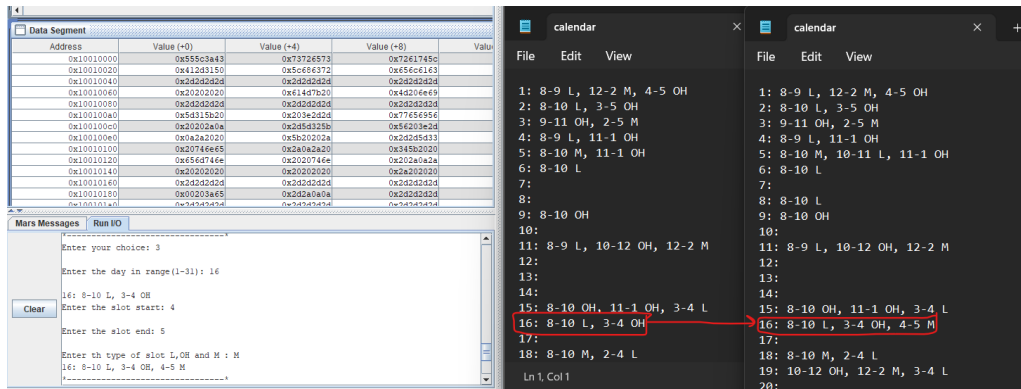
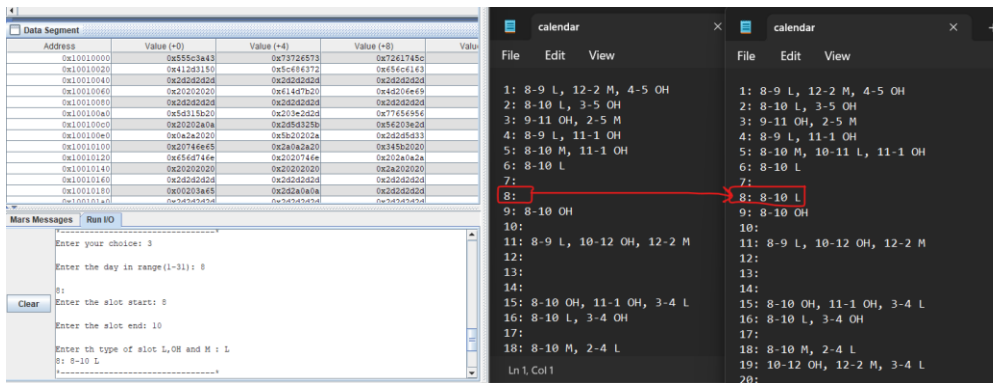
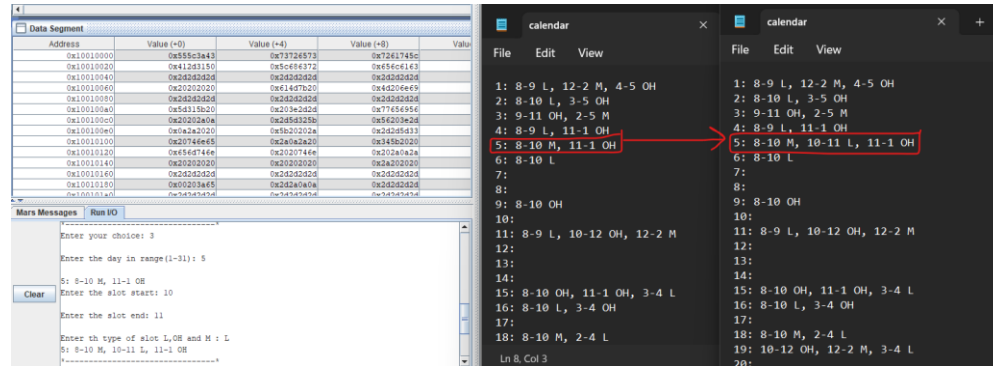
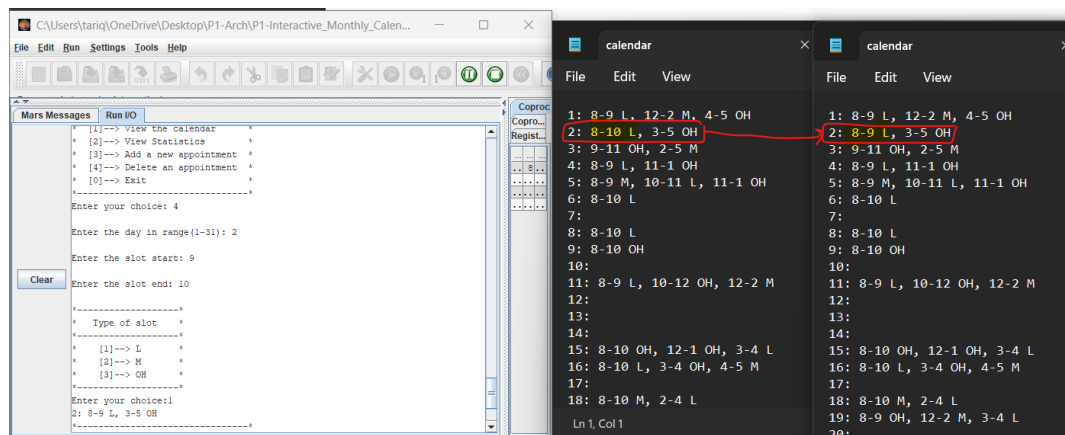
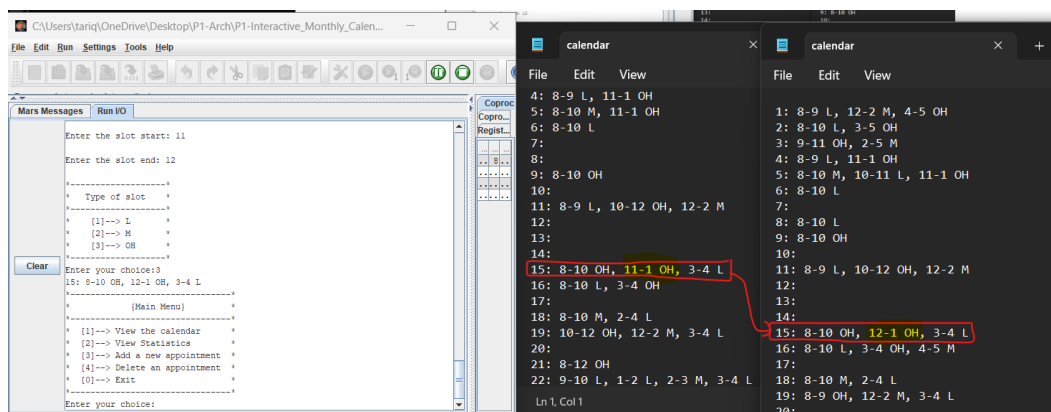
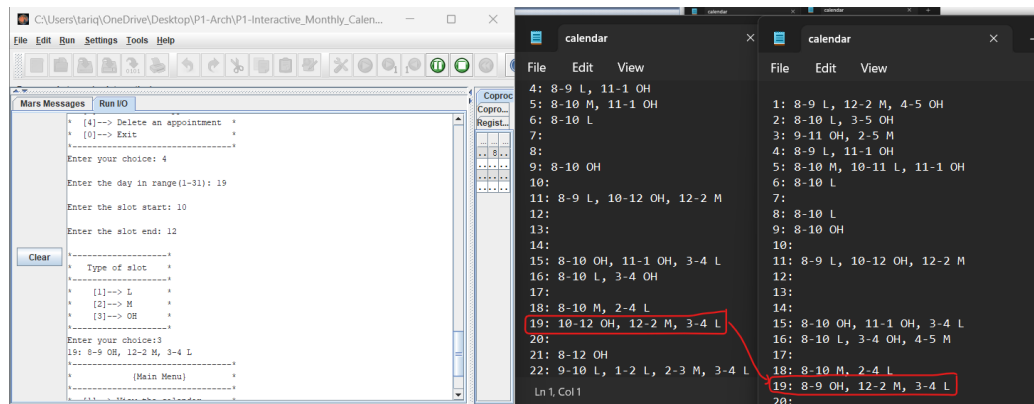


Figure 5-15: View Statistics after edit

## 5.4. Add a new appointment



## 5.5. Delete an appointment



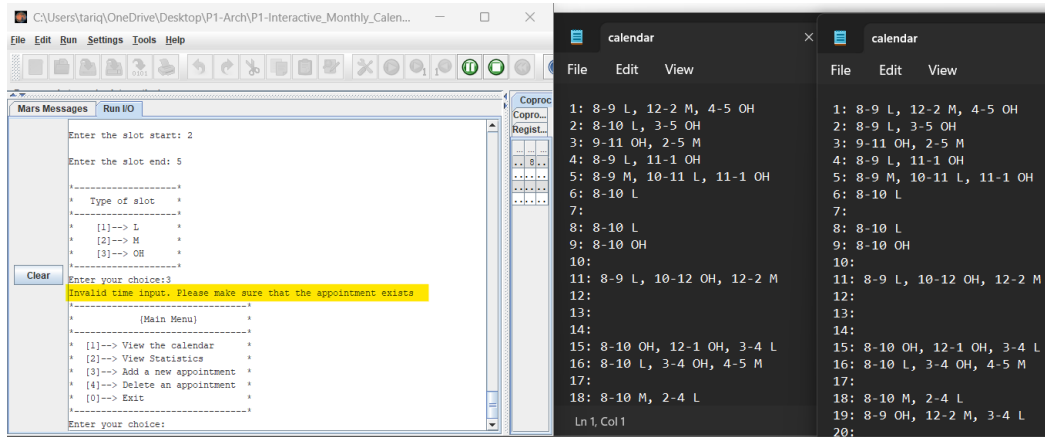


Figure 5-23: Delete from slot (invalid input)

## 5.6. Exit

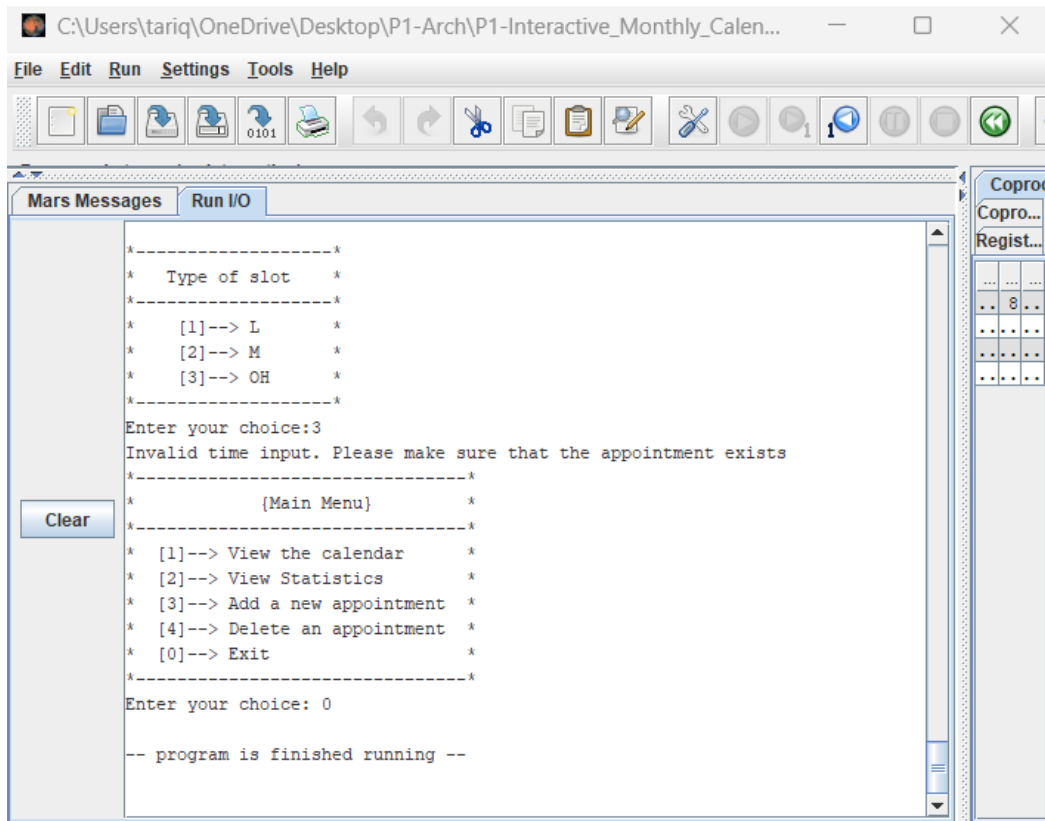


Figure 5-24: Exit from prgram



## 6. Appendix

1: 8-9L, 11-20H, 4-5M

S/E=0  
 $\begin{matrix} \text{F/L}=0 & (\text{S}) & \text{Start} \rightarrow \text{char/x} \\ \text{F/L}=1 & (\text{S}) & \text{Start} \rightarrow \text{Start/skip} \\ \text{F/L}=0 & (\text{S}) & \text{end} \rightarrow \text{char} \\ \text{F/L}=1 & (\text{S}) & \text{end} \rightarrow \text{char} \end{matrix}$

S/E=1  
 $\begin{matrix} \text{F/L}=0 & (\text{S}) & \text{end} \rightarrow \text{char} \\ \text{F/L}=1 & (\text{S}) & \text{end} \rightarrow \text{char} \end{matrix}$

8-10	(8-10) ✓	✓ pass	Char/char/X	99
11-2	(11-2) ✓	✓ pass	skip/skip/char/char/X	112
12-1	(12-1) ✓	✓ pass	skip/skip/skip/start/end/X	12
10-11	(10-11) ✓	✓ pass	skip/skip/skip/X	11 11
9-11	(9-11) ✓	✗	skip/start/char/end/X	11 11
9-12	(9-12) ✓	✓ pass	skip/start/char/end/X	
2-4		✓ pass	skip/skip/skip/start/char/end/X	
12-3		✗	skip/skip/skip/start/X	
8-12		✓ pass	char/char/char/end/X	
11-5		✓		

8-5 Time

8, 9, 10, 11, 12

1, 2, 3, 4, 5 (+12)

Figure 6-1: Design view algorithm -1

1: 8-9L, 11-20H, 4-5M

	Test 1	Test 2	Test 3
8-10 → 8 9 X	8 9 10 10	8 9 10 10	
11-2 → skip skip 11 2 X	11 14 14	11 11 11 14	
12-1 → skip skip 12 1 X	16	12 12 16	12 13
10-11 → skip skip 10 11 X	11 11 ✓	10 10 11 11	11 11
9-11 → skip 9 11 11 X	11 11 ✓	9 11 11	9 11 11
9-12 → skip 9 11 12 X	11 12 ✓	9 11 12	9 9 12

Figure 6-2: Design view algorithm -2

22 <sup>o</sup> 8-10 L, 11-2 L, 2-3 M, 4-5 L				
Input	Check	Pass to Delete?	Delete	True?
8-9 L	c=1 R/W=0	✓	9-10 L	✓
12-2 L	c=2 R/W=0	✓	11-12 L	✓
2-3 M	c=1 R/W=0	✓	* * * * *	✓
4-5 M	c=1 R/W=1	X	X	✓
10-11 L	c=3 R/W=1	X	X	✓
2-4 M	c=1 R/W=1	X	X	✓

Check	Delete
$\begin{aligned} & \text{C/L} = 0 \leq \text{start} \rightarrow \text{Delete} / \text{R/W} = 1 \\ & \text{Skip} \\ & \text{C/L} = 1 \end{aligned}$	$\begin{aligned} & \text{C/L} = 0 = \text{start} \rightarrow \text{***} / \text{char} = \text{start} \\ & \text{C/L} = 1 \end{aligned}$
$\begin{aligned} & \text{C/L} = 0 \\ & \text{C/L} = 1 \geq \text{end} \rightarrow \text{R/W} = 0 \\ & \text{go start} / \text{R/W} = 1 \\ & \text{Skip} \end{aligned}$	$\begin{aligned} & \text{C/L} = 0 \\ & \text{C/L} = 1 = \text{end} \rightarrow \text{go start} / \text{end} = \text{char} \end{aligned}$
$\begin{aligned} & \text{R/W} \rightarrow 0/1 \\ & \text{Count} \rightarrow 0 \end{aligned}$	

Figure 6-3: Design delete algorithm