



Universiti Tunku Abdul Rahman
Faculty of Information and Communication Technology
UCCD1004 Programming Concepts And Practises

Assignment 2
Group Number: 58

No	Name	Student ID	Programme Name
1.	Sam Jian Him	1903626	BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) COMMUNICATIONS AND NETWORKING
2.	Calvin Lau Kai Wee	1903711	BACHELOR OF COMPUTER SCIENCE

Email Address: jianhim01@utar.my
calvinlaukaiwee2001@utar.my
Date of Submission 7/9/2020

Part A - The Assignment Objective

Write a C++ program which enable an admin to book a ZOOM account (on behalf of lecturers) for a particular course to conduct online classes.

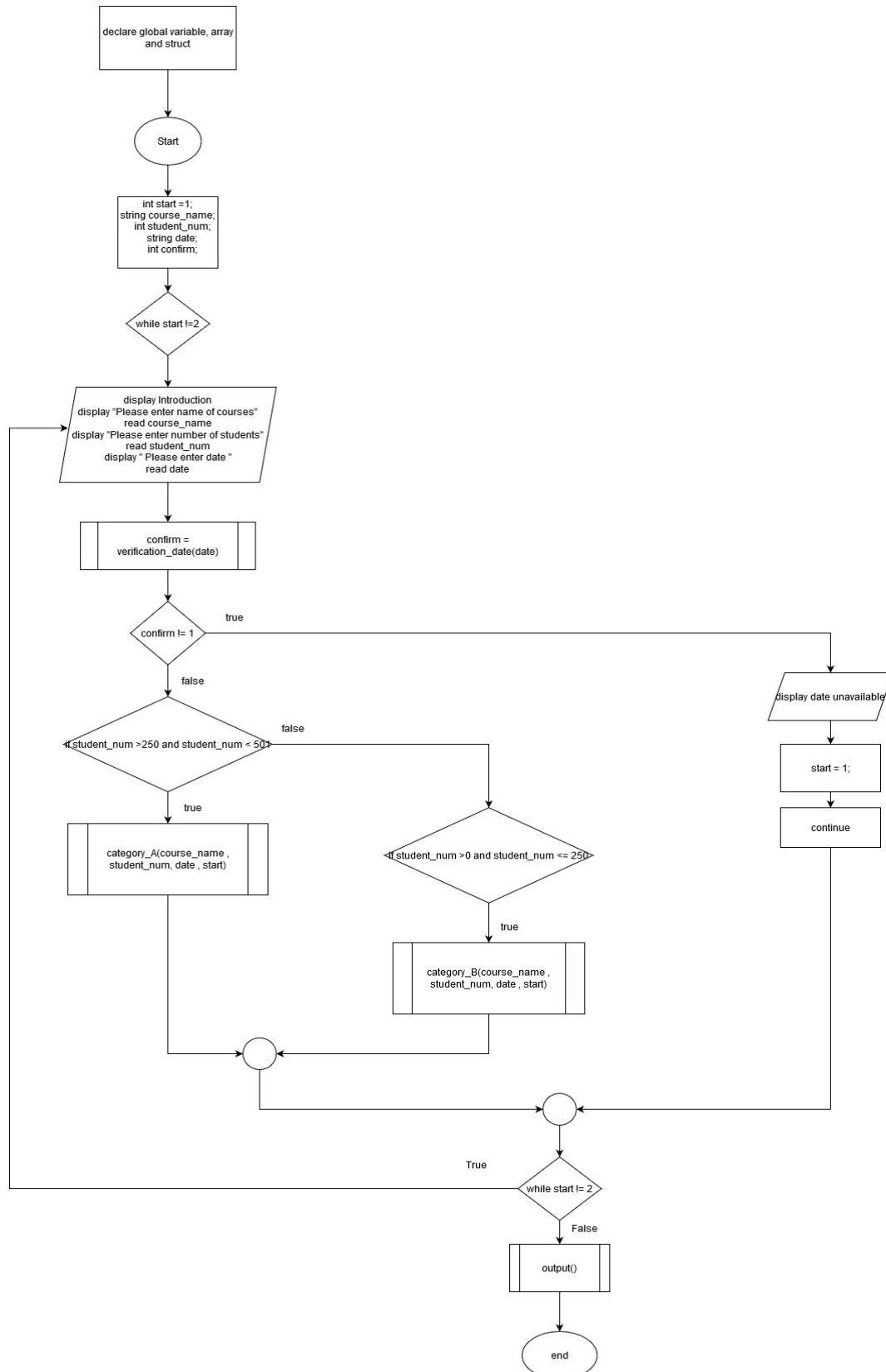
Part B – Task Division

Name	Task
Sam Jian Him (Member 1)	Add and Search Modules
Calvin Lau Kai Wee (Member 2)	Delete and Update Modules

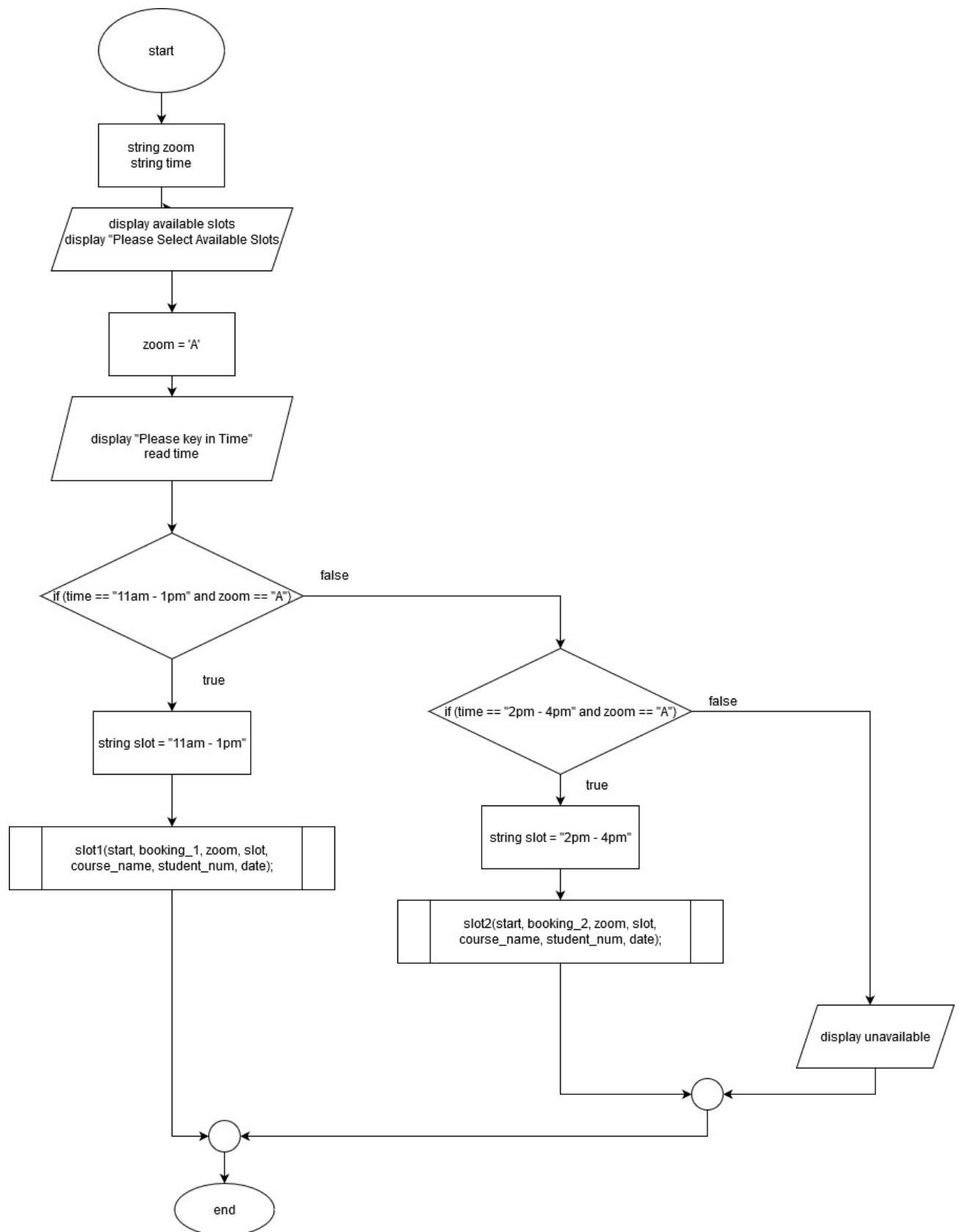
Part C – Flowchart

Flowchart

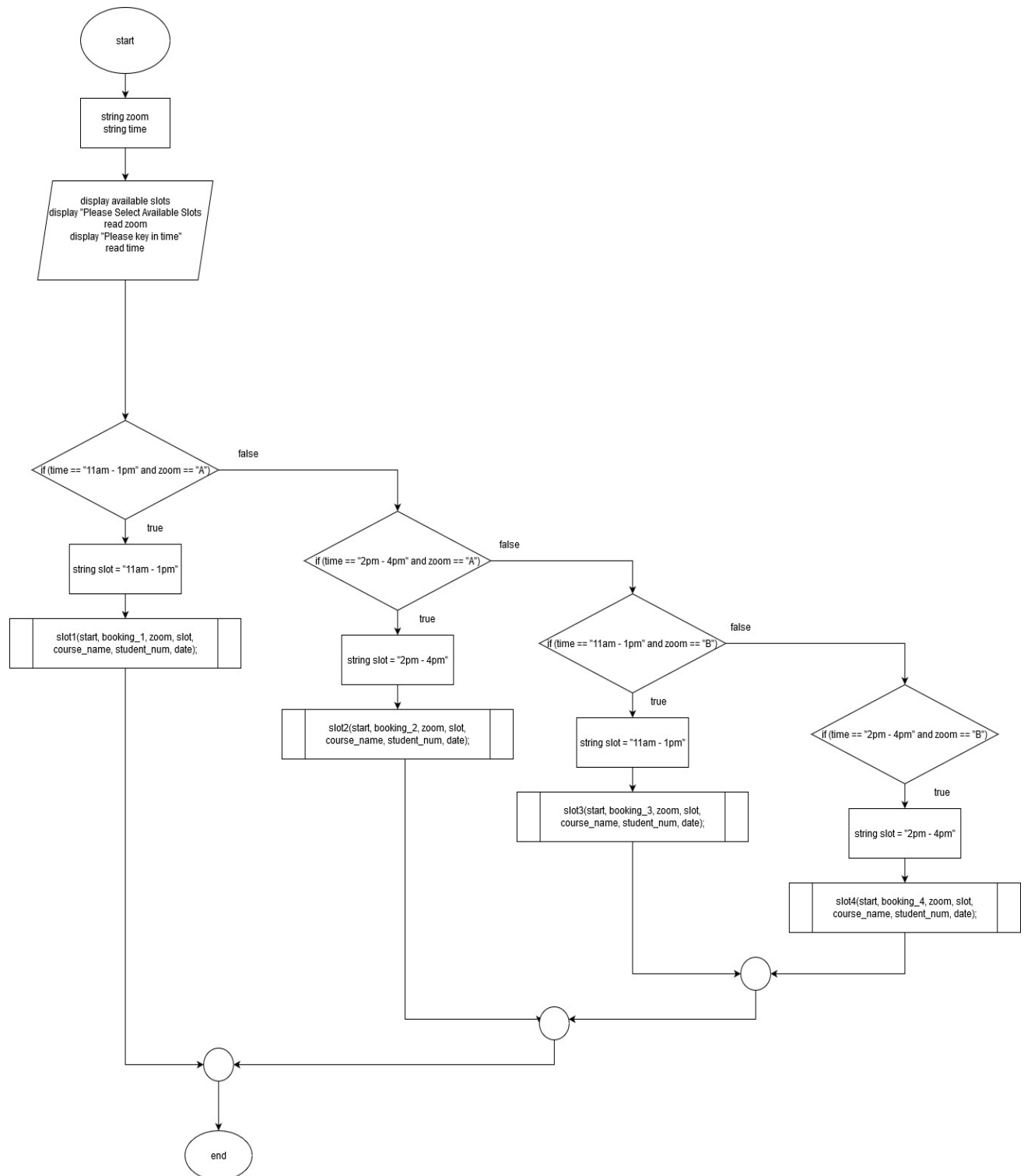
1.) Int Main Function



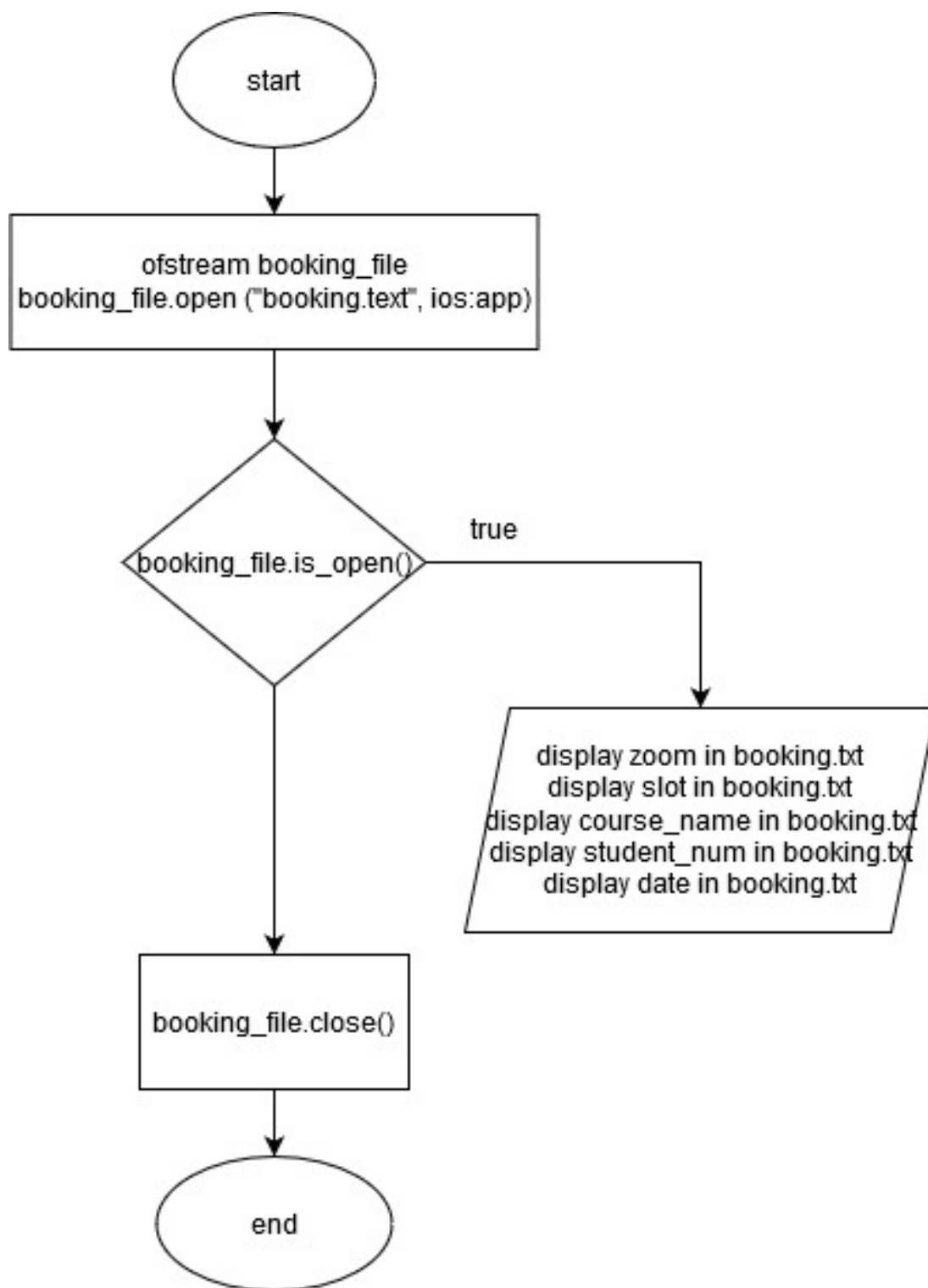
2.) Category_A Function



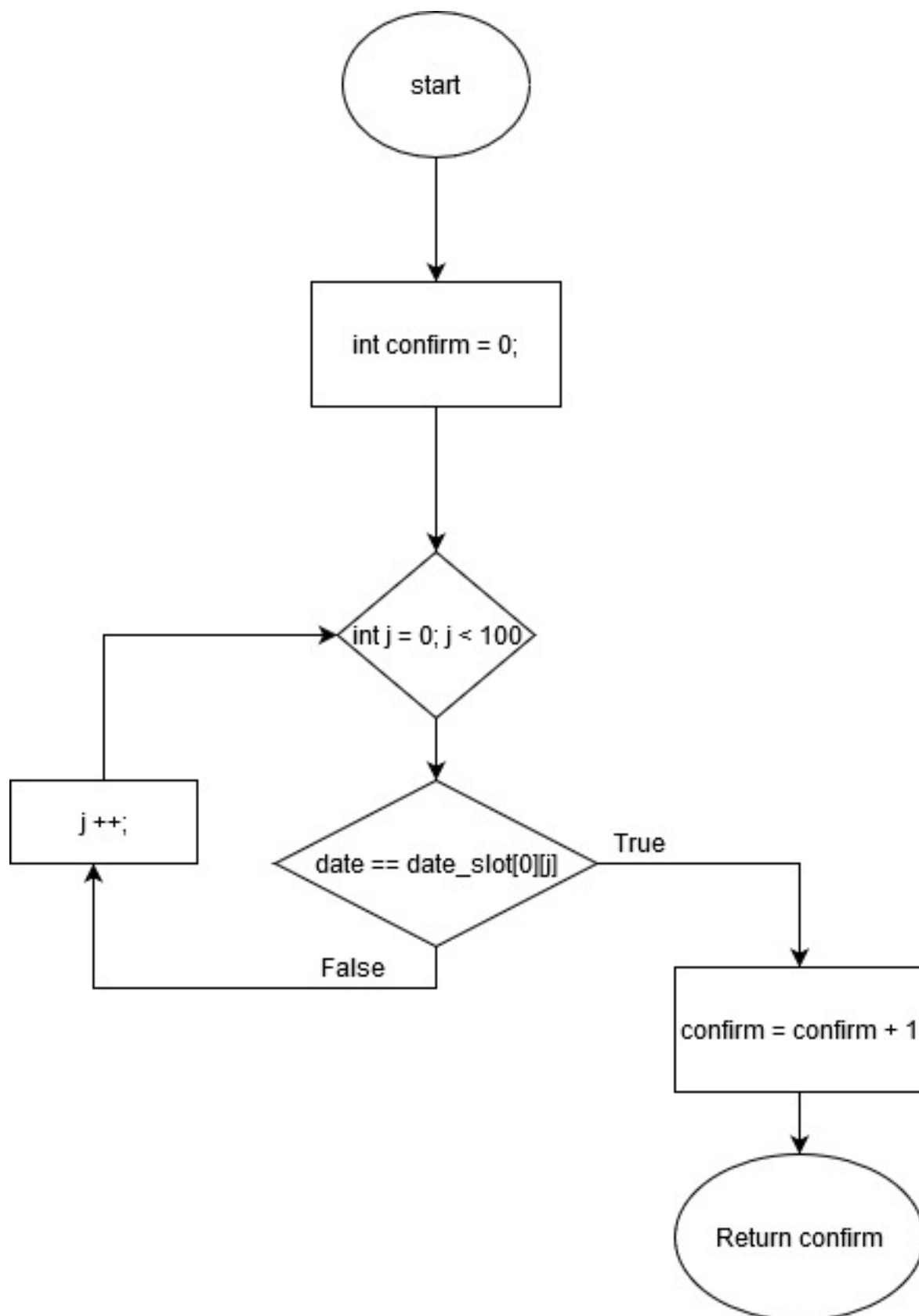
3.) Category_B Function



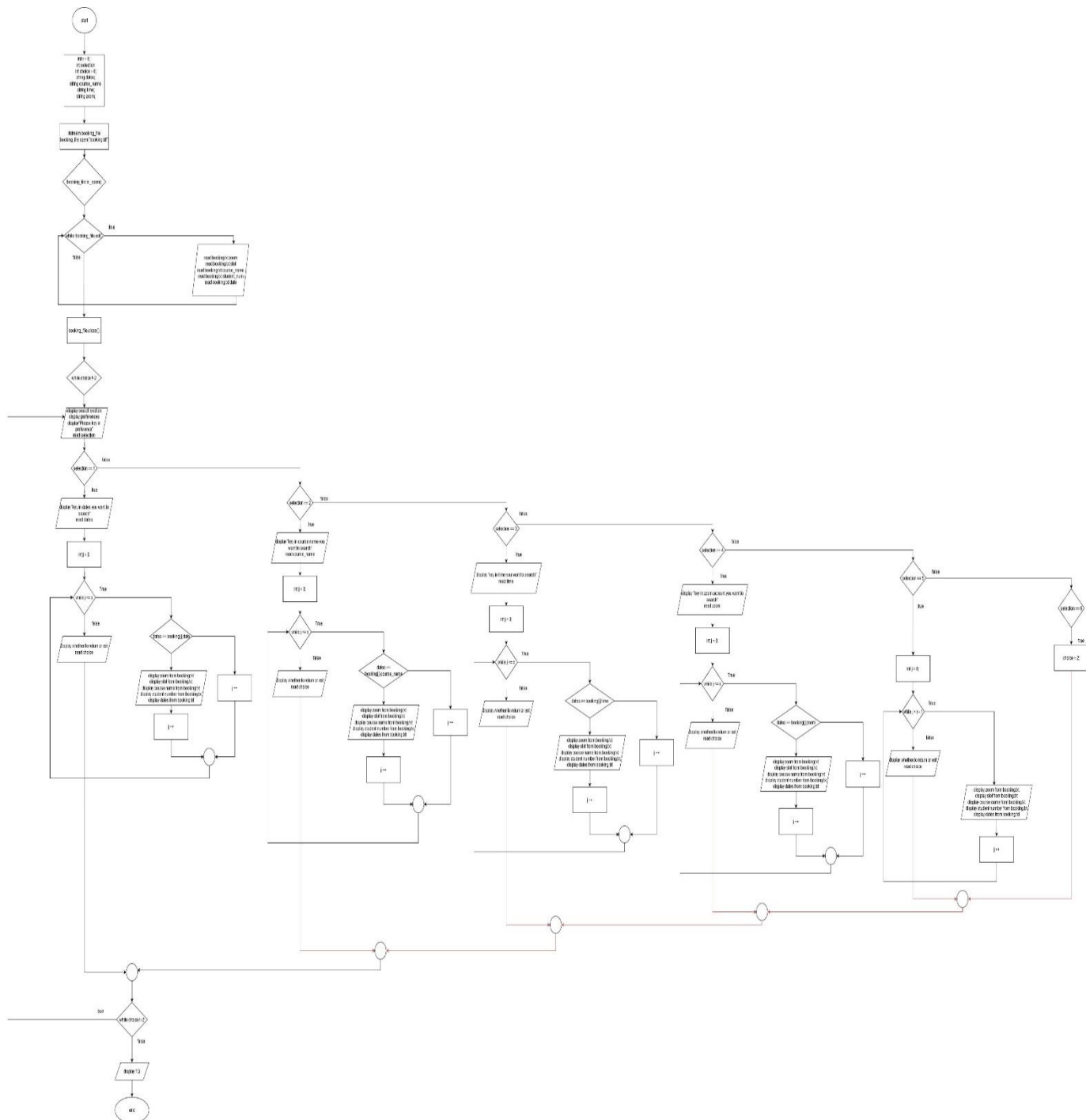
4.) Record Function



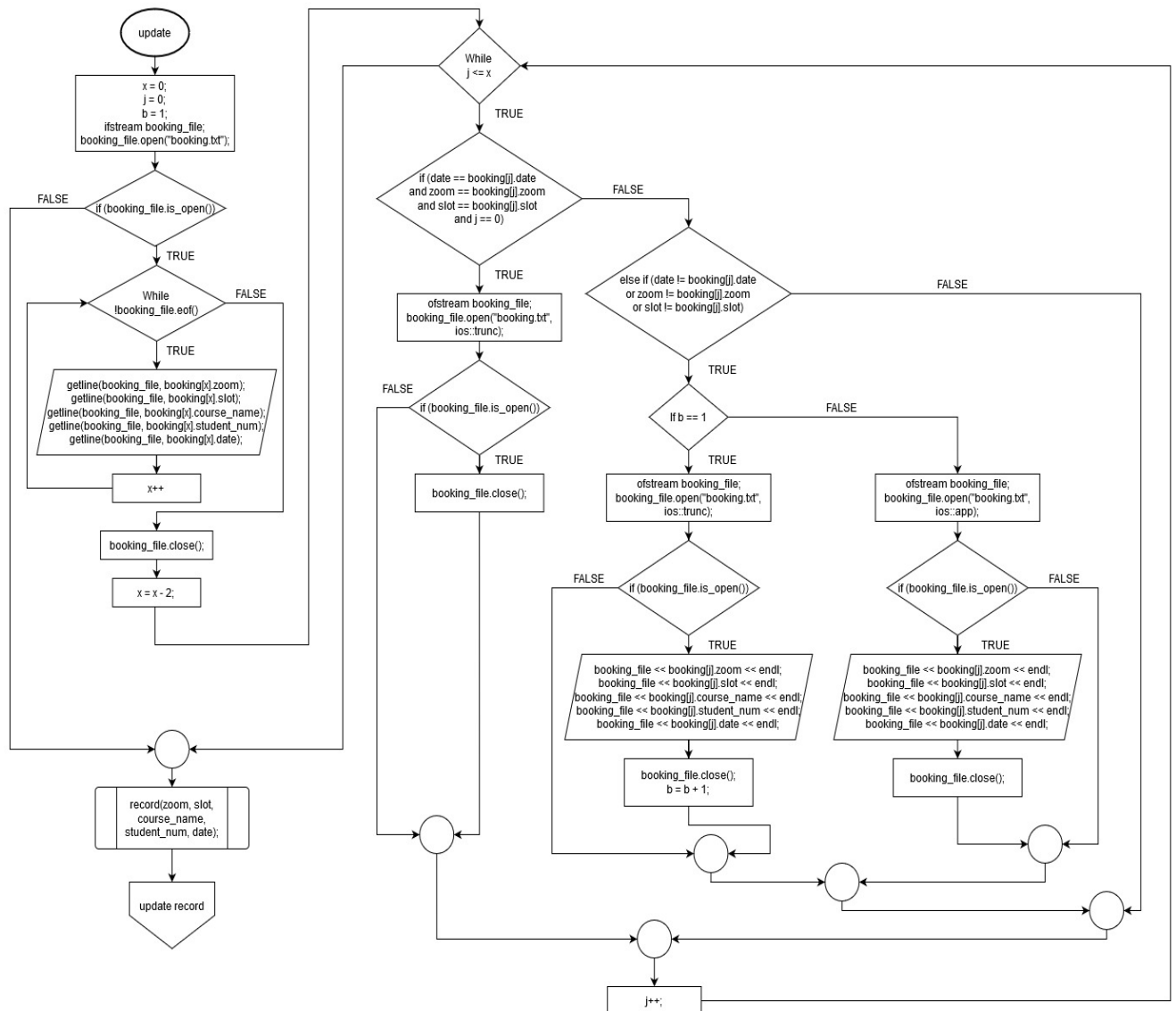
5.) Verification date Function



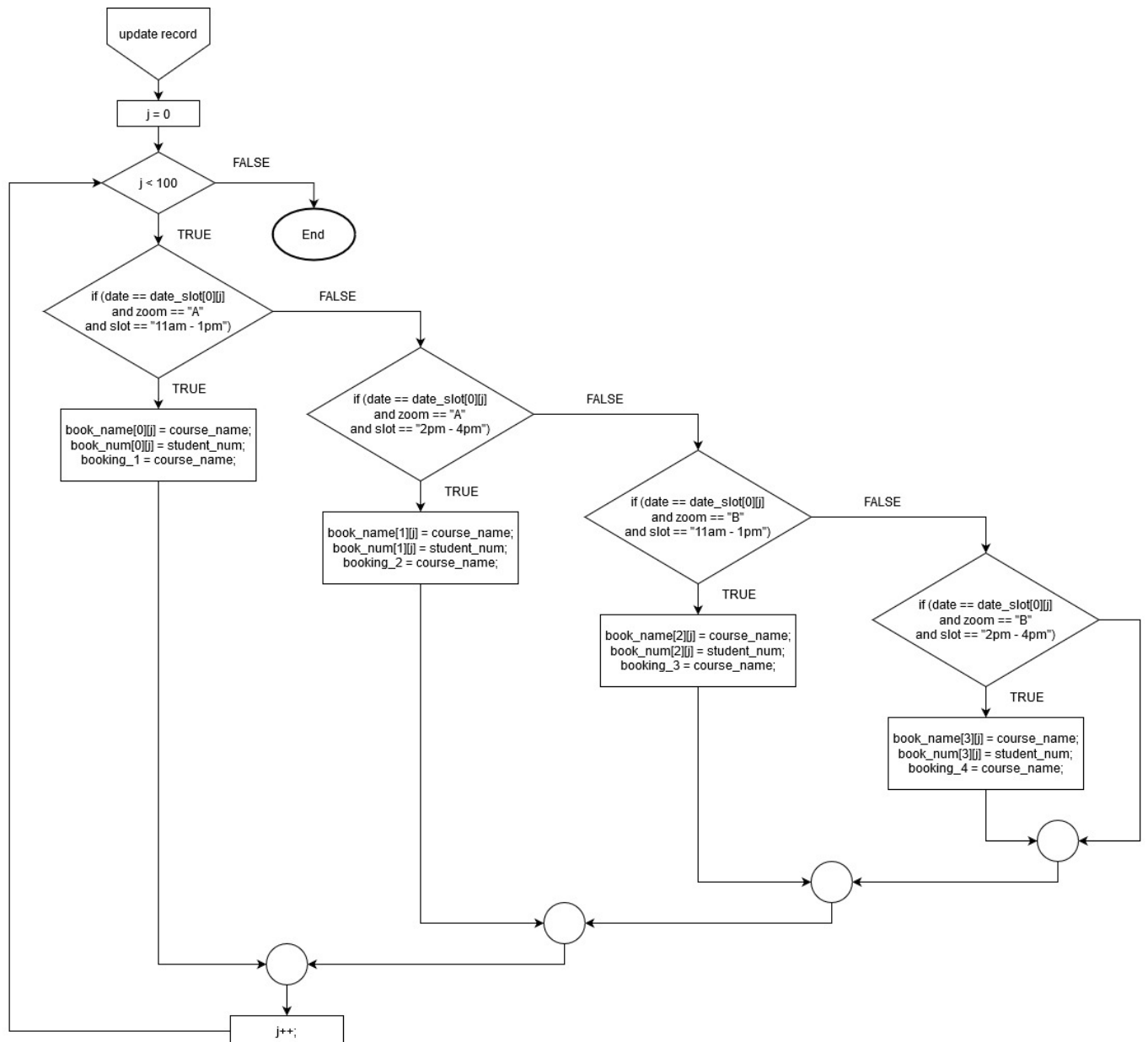
6.) Output Function



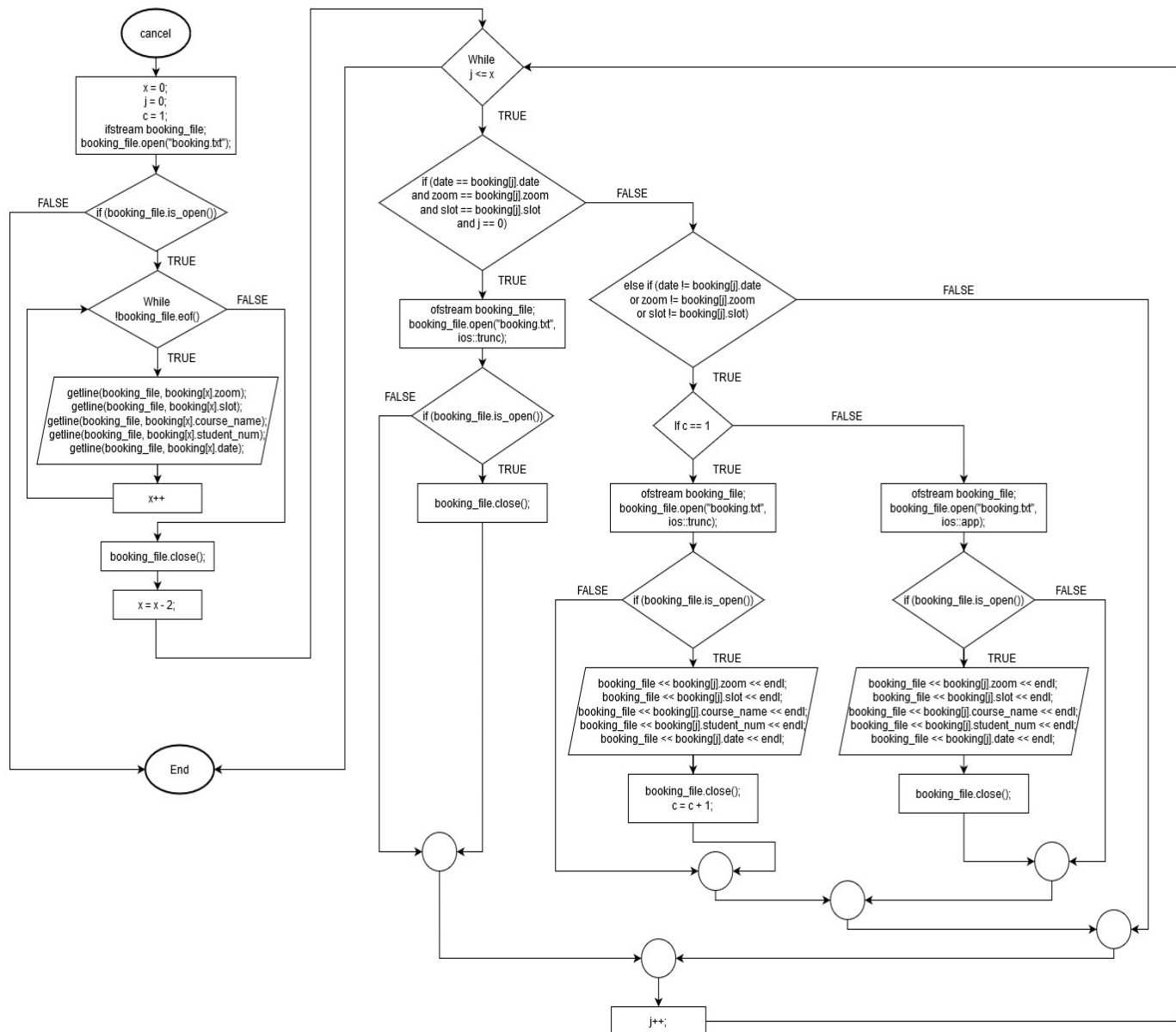
7.) Update Function (Part 1)



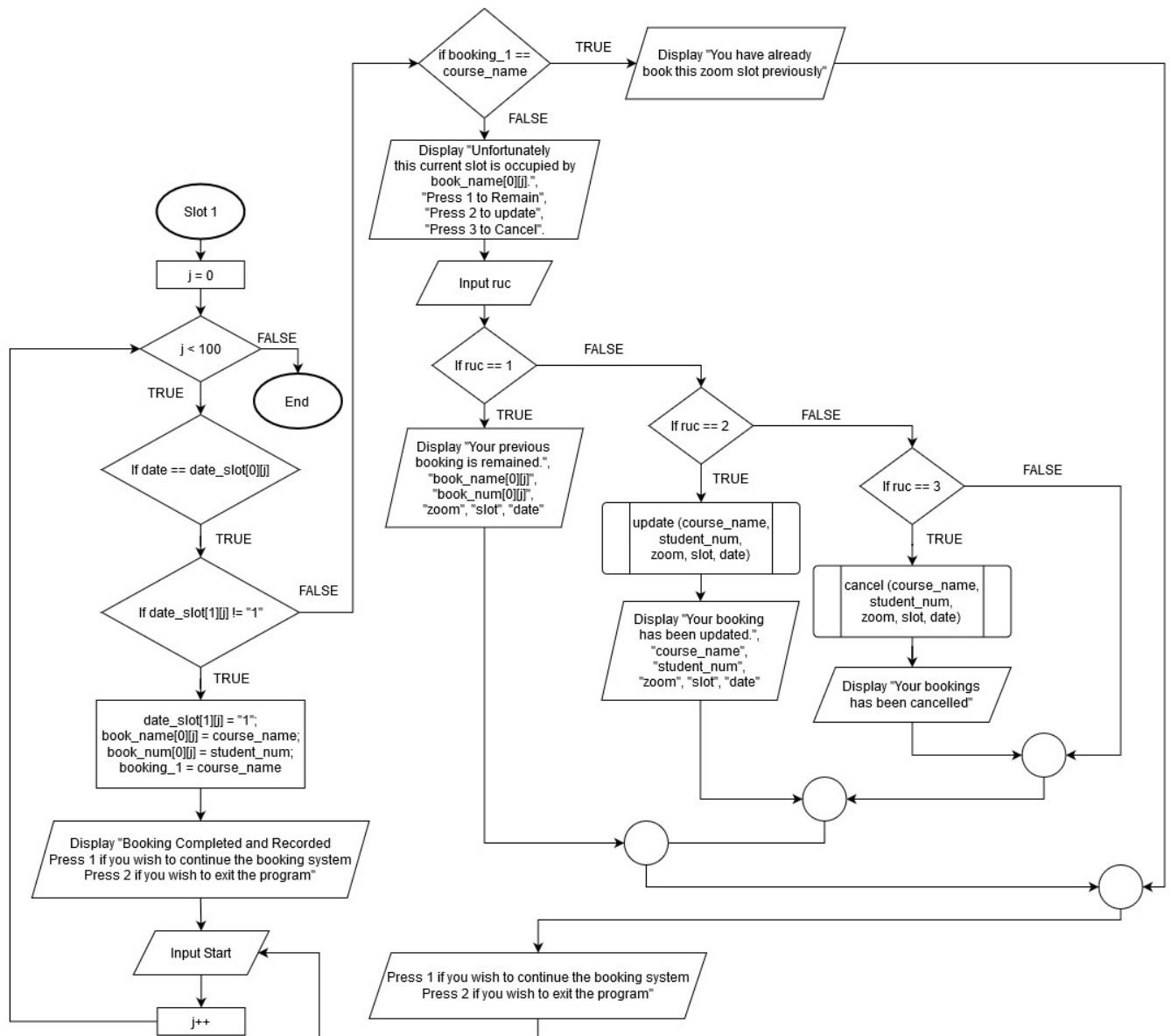
8.) Update Function (Part 2)



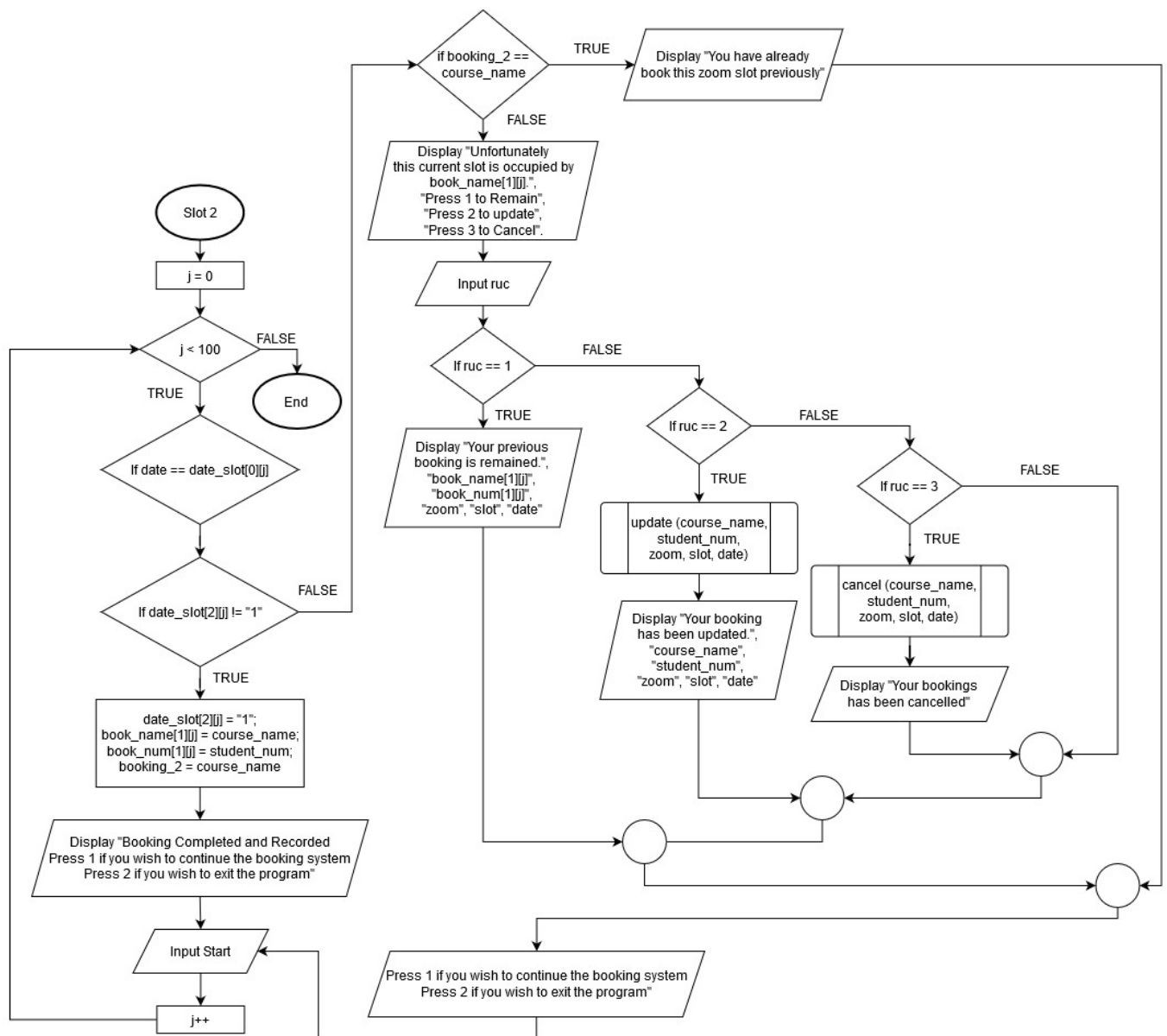
9.) Cancel Function



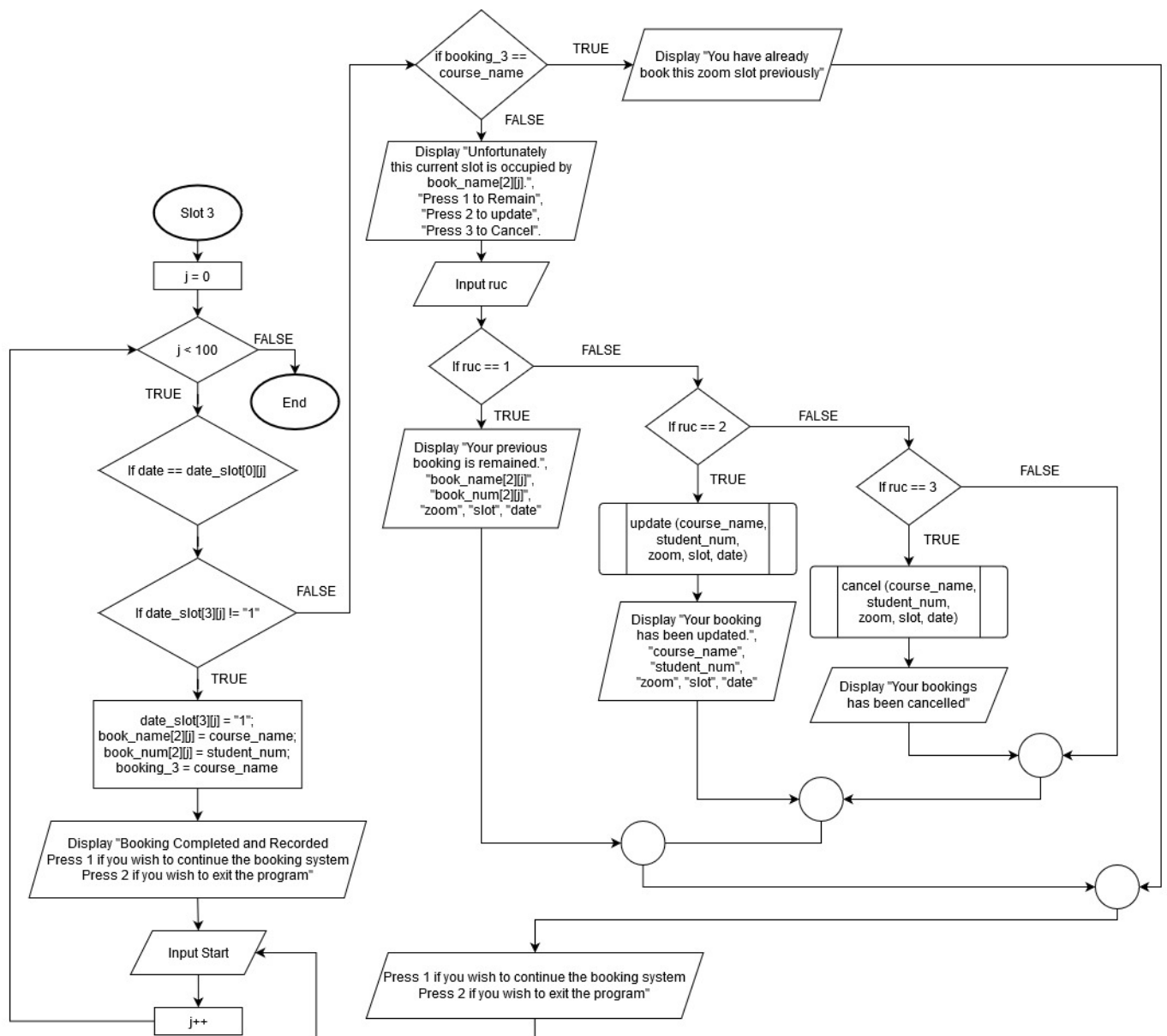
10.) Slot_1 Function



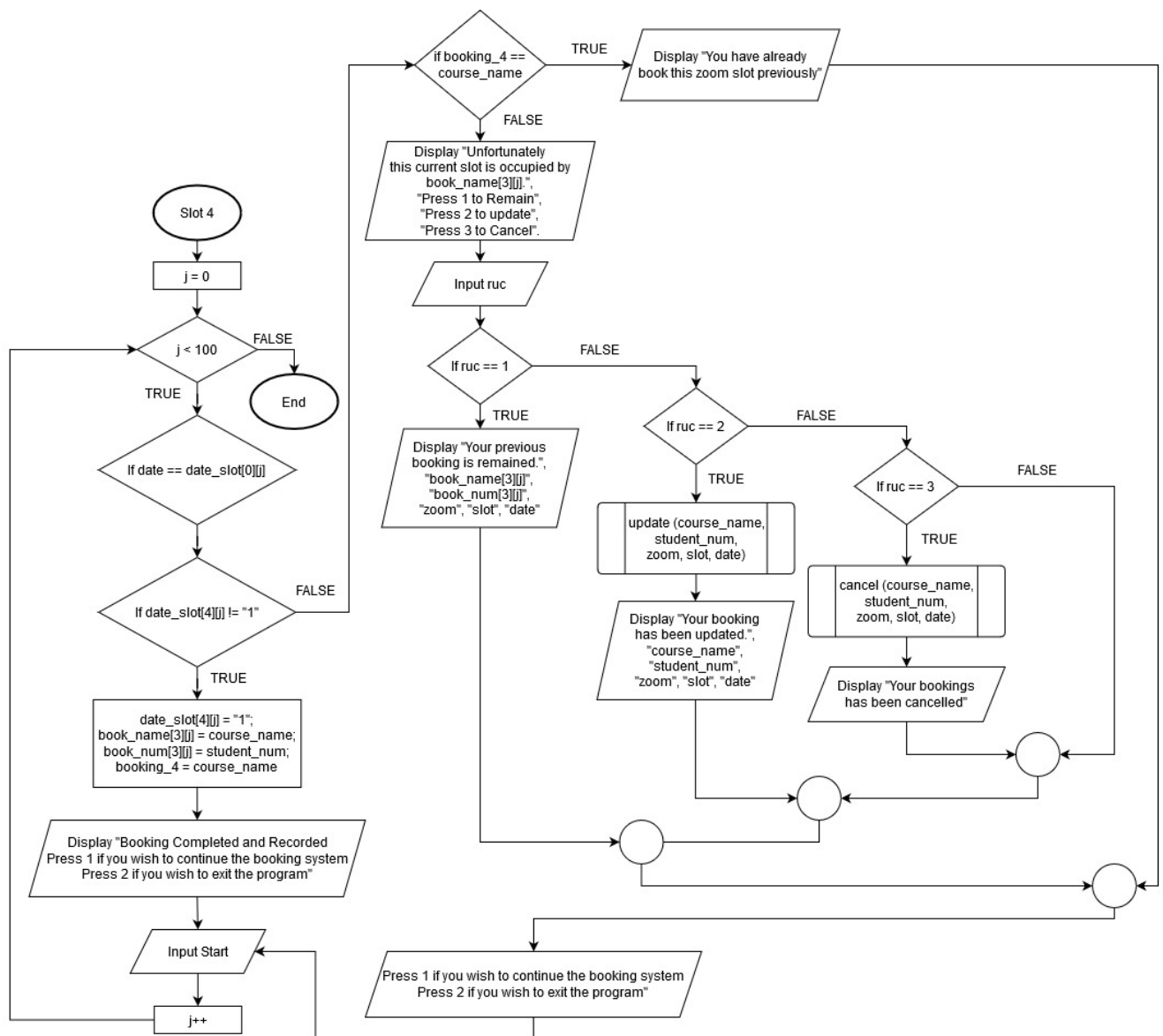
11.) Slot_2 Function



12.) Slot_3 Function



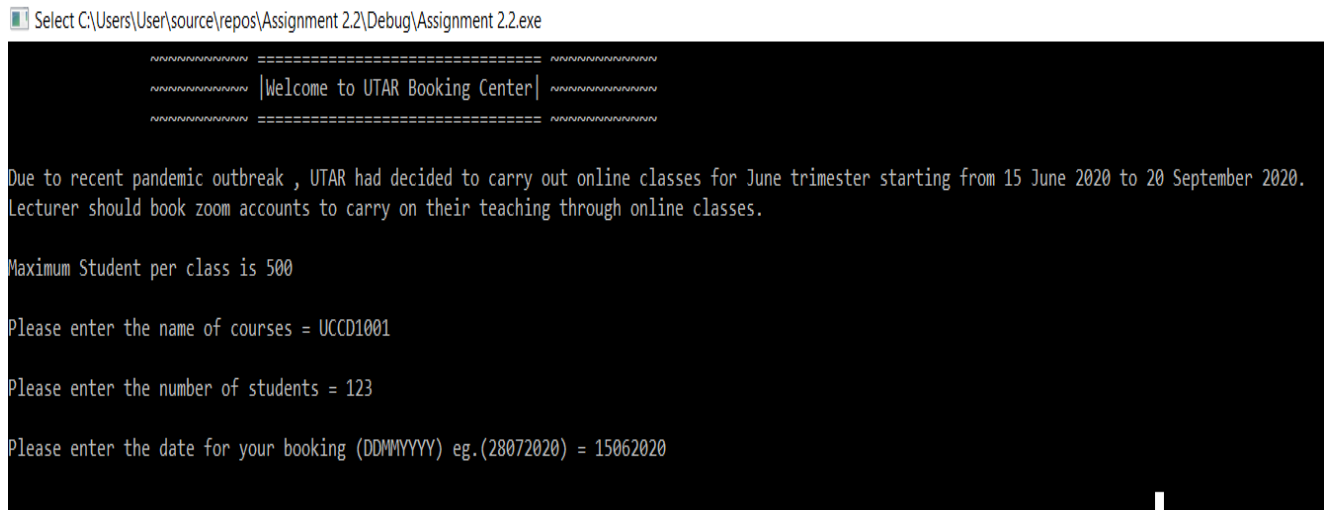
13.) Slot_4 Function



Part D – Test Cases

Test Case 1: The First Recording

A.) Key in course name, number of student and date.



```
Select C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

=====
|Welcome to UTAR Booking Center|
=====

Due to recent pandemic outbreak , UTAR had decided to carry out online classes for June trimester starting from 15 June 2020 to 20 September 2020.
Lecturer should book zoom accounts to carry on their teaching through online classes.

Maximum Student per class is 500

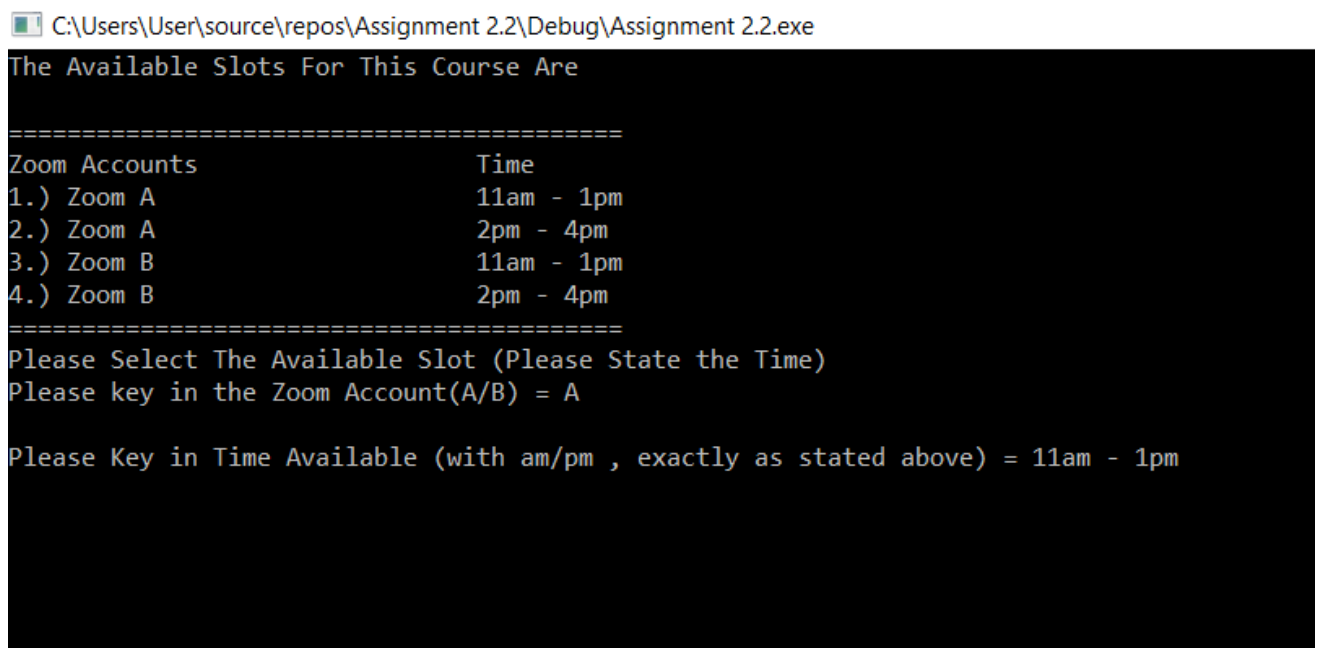
Please enter the name of courses = UCCD1001

Please enter the number of students = 123

Please enter the date for your booking (DDMMYYYY) eg.(28072020) = 15062020
```

Image 1.1 – The Main Interface

B.) Key in the zoom account and time slots.



```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

The Available Slots For This Course Are


=====
Zoom Accounts          Time
1.) Zoom A             11am - 1pm
2.) Zoom A             2pm - 4pm
3.) Zoom B             11am - 1pm
4.) Zoom B             2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
```

Image 1.2 – The Time Slot / Zoom Account Menu

C.) Booking Completed and Recorded

 C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```
The Available Slots For This Course Are

=====
Zoom Accounts          Time
1.) Zoom A            11am - 1pm
2.) Zoom A            2pm - 4pm
3.) Zoom B            11am - 1pm
4.) Zoom B            2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Booking Completed and Recorded

Thank you for booking!
-----
Your Booking are listed below:

Course code and name: UCCD1001
Number of Students: 123
Zoom Class: A
Time: 11am - 1pm
Date: 15062020

Do you now wish to continue to book or exit the program?
Press 1 if you wish to continue the booking system
Press 2 if you wish to exit the program
Your pick is = 1
```

Image 1.3 – The Recording Process

Test Case 2: Trying The Update Option

A.) Key in the same time and date as test case 1 to make the update option pop up

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```

=====
|Welcome to UTAR Booking Center|
=====

Due to recent pandemic outbreak , UTAR had decided to carry out online classes for June trimester starting from 15 June 2020 to 20 September 2020.
Lecturer should book zoom accounts to carry on their teaching through online classes.

Maximum Student per class is 500

Please enter the name of courses = UCCD1002

Please enter the number of students = 32

Please enter the date for your booking (DDMMYYYY) eg.(28072020) = 15062020
```

Image 2.1 – The Main Interface

B.) Key in the same slots and zoom accounts as test case 1

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```

The Available Slots For This Course Are

=====
Zoom Accounts      Time
1.) Zoom A        11am - 1pm
2.) Zoom A        2pm - 4pm
3.) Zoom B        11am - 1pm
4.) Zoom B        2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm_
```

Image 2.2 – The Time Slot / Zoom Account Menu

C.) Key in the Update Option (Selection = 2)

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
The Available Slots For This Course Are

=====
Zoom Accounts      Time
1.) Zoom A        11am - 1pm
2.) Zoom A        2pm - 4pm
3.) Zoom B        11am - 1pm
4.) Zoom B        2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Unfortunately this current slot is occupied by UCCD1001
Press 1 to Remain (There will be no changes, everything remain unchange)
Press 2 to Update (Replace the current data with the previous data)
Press 3 to Cancel (Remove the current data and previous data)
Your Selection Is = 2
```

Image 2.3 The Remain/Update/Cancel Options

D.) Booking Updated

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
The Available Slots For This Course Are

=====
Zoom Accounts      Time
1.) Zoom A        11am - 1pm
2.) Zoom A        2pm - 4pm
3.) Zoom B        11am - 1pm
4.) Zoom B        2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Unfortunately this current slot is occupied by UCCD1001
Press 1 to Remain (There will be no changes, everything remain unchange)
Press 2 to Update (Replace the current data with the previous data)
Press 3 to Cancel (Remove the current data and previous data)
Your Selection Is = 2

Your booking has been updated
-----
Your Booking are listed below:

Course code and name: UCCD1002
Number of Students: 32
Zoom Class: A
Time: 11am - 1pm
Date: 15062020

Do you now wish to continue to book or exit the program?
Press 1 if you wish to continue the booking system
Press 2 if you wish to exit the program
Your pick is =
```

Image 2.4 – The Updating Process

Test Case 3: Trying the cancel option

A.) Key in the same time and date as test case 2 to make the cancel option pop up

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```

=====
|Welcome to UTAR Booking Center|
=====

Due to recent pandemic outbreak , UTAR had decided to carry out online classes for June trimester starting from 15 June 2020 to 20 September 2020.
Lecturer should book zoom accounts to carry on their teaching through online classes.

Maximum Student per class is 500

Please enter the name of courses = UCCD1003

Please enter the number of students = 300

Please enter the date for your booking (DDMMYYYY) eg.(28072020) = 15062020
```

Image 3.1 - The Main Interface

B.) Key in the same slots and zoom accounts as test case 2

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```

The Available Slots For This Course Are

=====
Zoom Accounts          Time
1.) Zoom A             11am - 1pm
2.) Zoom A             2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm_
```

Image 3.2 – The Time Slot / Zoom Account Menu

C.) Key in the Cancel Option (Selection = 3)

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```
The Available Slots For This Course Are

=====
Zoom Accounts      Time
1.) Zoom A        11am - 1pm
2.) Zoom A        2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Unfortunately this current slot is occupied by UCCD1002
Press 1 to Remain (There will be no changes, everything remain unchange)
Press 2 to Update (Replace the current data with the previous data)
Press 3 to Cancel (Remove the current data and previous data)
Your Selection Is = 3_
```

Image 3.3 The Remain/Update/Cancel Options

D.) Booking Removed

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```
The Available Slots For This Course Are

=====
Zoom Accounts      Time
1.) Zoom A        11am - 1pm
2.) Zoom A        2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Unfortunately this current slot is occupied by UCCD1002
Press 1 to Remain (There will be no changes, everything remain unchange)
Press 2 to Update (Replace the current data with the previous data)
Press 3 to Cancel (Remove the current data and previous data)
Your Selection Is = 3

Your bookings has been cancelled
-----
Your Booking are listed below:

Course code and name: -
Number of Students: -
Zoom Class: -
Time: -
Date: -

Do you now wish to continue to book or exit the program?
Press 1 if you wish to continue the booking system
Press 2 if you wish to exit the program
Your pick is = _
```

Image 3.4 – The Cancelling Process

Test Case 4: Trying The Remain Option

A.) Key in a new time and slot for testing

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
The Available Slots For This Course Are
=====
Zoom Accounts      Time
1.) Zoom A         11am - 1pm
2.) Zoom A         2pm - 4pm
3.) Zoom B         11am - 1pm
4.) Zoom B         2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Booking Completed and Recorded

Thank you for booking!
-----
Your Booking are listed below:

Course code and name: UCCD1004
Number of Students: 35
Zoom Class: A
Time: 11am - 1pm
Date: 15062020

Do you now wish to continue to book or exit the program?
Press 1 if you wish to continue the booking system
Press 2 if you wish to exit the program
Your pick is =
```

Image 4.1 – The Record Process

B.) Key in the same time and date as image 4.1 to make the remain option pop up

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
=====
|Welcome to UTAR Booking Center|
=====

Due to recent pandemic outbreak , UTAR had decided to carry out online classes for June trimester starting from 15 June 2020 to 20 September 2020.
Lecturer should book zoom accounts to carry on their teaching through online classes.

Maximum Student per class is 500

Please enter the name of courses = UCCD1005

Please enter the number of students = 32

Please enter the date for your booking (DDMMYYYY) eg.(28072020) = 15062020_
```

Image 4.2 - The Main Interface

C.) Key in the same slots and zoom accounts as above image 4.1

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
The Available Slots For This Course Are
=====
Zoom Accounts          Time
1.) Zoom A            11am - 1pm
2.) Zoom A            2pm - 4pm
3.) Zoom B            11am - 1pm
4.) Zoom B            2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
```

Image 4.3 – The Time Slot / Zoom Account Menu

D.) Key in the Cancel Option (Selection = 3)

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
The Available Slots For This Course Are
=====
Zoom Accounts          Time
1.) Zoom A            11am - 1pm
2.) Zoom A            2pm - 4pm
3.) Zoom B            11am - 1pm
4.) Zoom B            2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Unfortunately this current slot is occupied by UCCD1004
Press 1 to Remain (There will be no changes, everything remain unchange)
Press 2 to Update (Replace the current data with the previous data)
Press 3 to Cancel (Remove the current data and previous data)
Your Selection Is = 1_
```

Image 4.4 The Remain/Update/Cancel Options

E.) Booking Remained (same as image 4.1)

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```
The Available Slots For This Course Are

=====
Zoom Accounts          Time
1.) Zoom A             11am - 1pm
2.) Zoom A             2pm - 4pm
3.) Zoom B             11am - 1pm
4.) Zoom B             2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
Unfortunately this current slot is occupied by UCCD1004
Press 1 to Remain (There will be no changes, everything remain unchange)
Press 2 to Update (Replace the current data with the previous data)
Press 3 to Cancel (Remove the current data and previous data)
Your Selection Is = 1

Your previous booking is remained.
-----
Your Booking are listed below:

Course code and name: UCCD1004
Number of Students: 35
Zoom Class: A
Time: 11am - 1pm
Date: 15062020

Do you now wish to continue to book or exit the program?
Press 1 if you wish to continue the booking system
Press 2 if you wish to exit the program
Your pick is =
```

Image 4.5 – The Remaining Process

Test Case 5: If we book the same slot and time with the same course name

A.) Key in the same time and date as image 4.1

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

=====
|Welcome to UTAR Booking Center|
=====

Due to recent pandemic outbreak , UTAR had decided to carry out online classes for June trimester starting from 15 June 2020 to 20 September 2020.
Lecturer should book zoom accounts to carry on their teaching through online classes.

Maximum Student per class is 500

Please enter the name of courses = UCCD1004

Please enter the number of students = 35

Please enter the date for your booking (DDMMYYYY) eg.(28072020) = 15062020
```

Image 4.1 - The Main Interface

B.) Key in the same slots and zoom accounts as above image 4.1

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

The Available Slots For This Course Are


=====
Zoom Accounts          Time
1.) Zoom A             11am - 1pm
2.) Zoom A             2pm - 4pm
3.) Zoom B             11am - 1pm
4.) Zoom B             2pm - 4pm
=====

Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm_
```

Image 5.2 – The Time Slot / Zoom Account Menu

- C.) A message will be sent to remind the user not to book the same slot with the same course

 C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```
The Available Slots For This Course Are

=====
Zoom Accounts      Time
1.) Zoom A        11am - 1pm
2.) Zoom A        2pm - 4pm
3.) Zoom B        11am - 1pm
4.) Zoom B        2pm - 4pm
=====
Please Select The Available Slot (Please State the Time)
Please key in the Zoom Account(A/B) = A

Please Key in Time Available (with am/pm , exactly as stated above) = 11am - 1pm
=====
You have already book this zoom slot previously
=====
```

Image 5.3 – The Reminder Message

Test Case 6: The Search Part

A.) Key in multiple courses with different date and slots

A 11am - 1pm UCCD1001 400 15062020
A 2pm - 4pm UCCD1002 300 15062020
B 11am - 1pm UCCD1003 200 28072020
B 2pm - 4pm UCCD1004 100 28072020

Table 6.1 – 4 sample for search section

B.) Key In Your Preference

C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe

```
*&?!  =====  !?#
#*@  |Search Engine|  @!%
%&&  =====  *$#

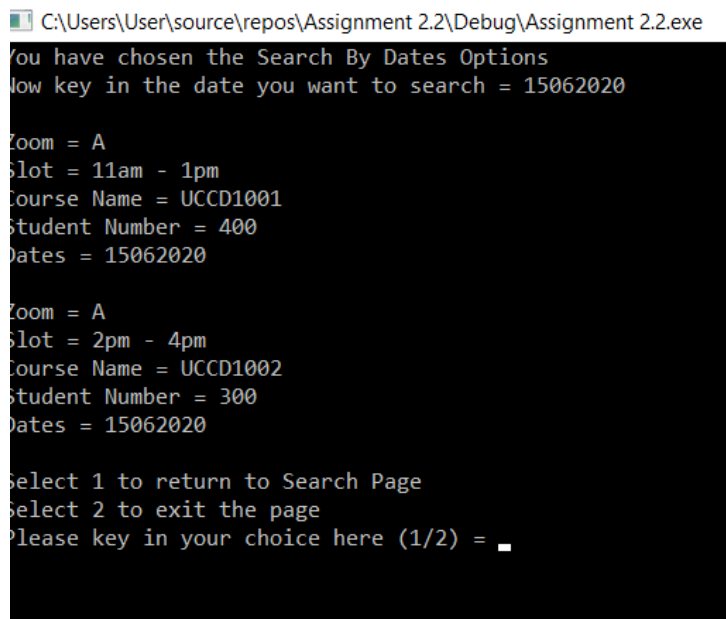
There are a few preferences for searching to ease your findings for bookings :D

||SEARCH PREFERENCES||

1.) Search By Dates
2.) Search By Course Name
3.) Search By Time Range
4.) Search By Zoom Account
5.) Search All (Every Booking)
6.) Exit Page
Please key in your preference (1/2/3/4/5/6) =
```

Image 6.2 – The Search Interface

C.) Date Preference for 15/06/2020 (Preference = 1)



```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
You have chosen the Search By Dates Options
Now key in the date you want to search = 15062020

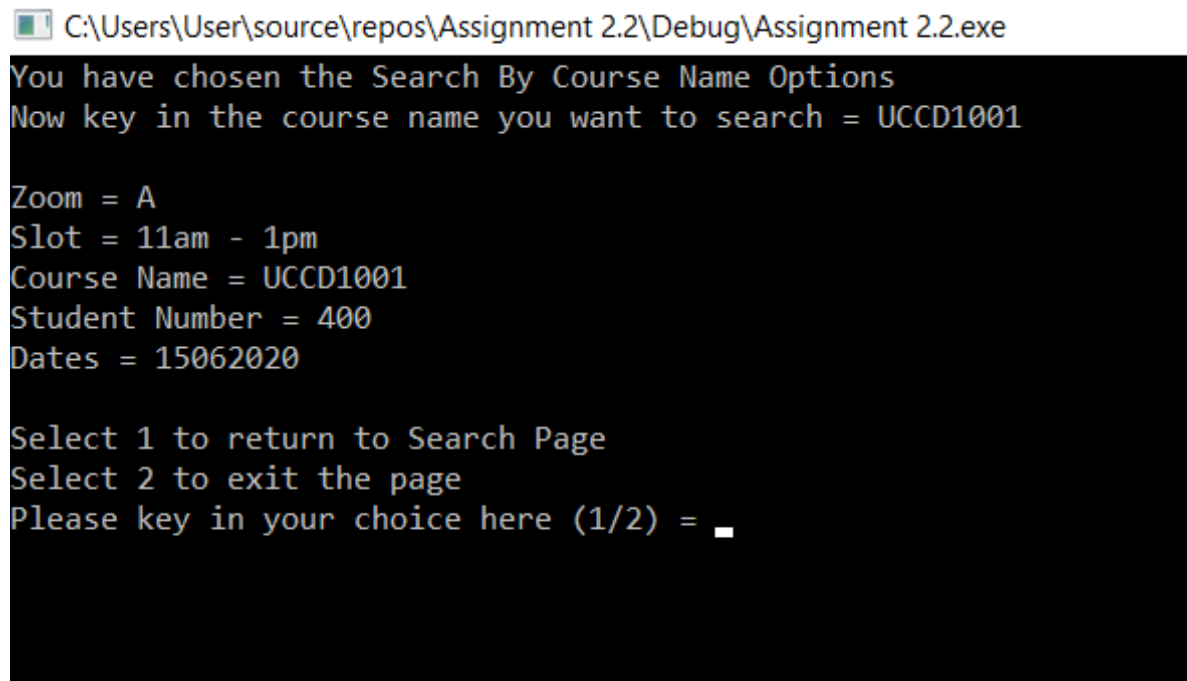
Zoom = A
Slot = 11am - 1pm
Course Name = UCCD1001
Student Number = 400
Dates = 15062020

Zoom = A
Slot = 2pm - 4pm
Course Name = UCCD1002
Student Number = 300
Dates = 15062020

Select 1 to return to Search Page
Select 2 to exit the page
Please key in your choice here (1/2) = █
```

Image 6.3 - Date Preference Interface

D.) Course Name Preference for UCCD1001 (Preference = 2)



```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
You have chosen the Search By Course Name Options
Now key in the course name you want to search = UCCD1001

Zoom = A
Slot = 11am - 1pm
Course Name = UCCD1001
Student Number = 400
Dates = 15062020

Select 1 to return to Search Page
Select 2 to exit the page
Please key in your choice here (1/2) = █
```

Image 6.4 – Course Name Preference Interface

E.) Time Range Preference for 11am – 1pm (Preference = 3)

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
You have chosen the Search By Time Options
Now key in the time range you want to search = 11am - 1pm

Zoom = A
Slot = 11am - 1pm
Course Name = UCCD1001
Student Number = 400
Dates = 15062020

Zoom = B
Slot = 11am - 1pm
Course Name = UCCD1003
Student Number = 200
Dates = 28072020

Select 1 to return to Search Page
Select 2 to exit the page
Please key in your choice here (1/2) = _
```

Image 6.5 – Time Range Preference Interface

F.) Zoom Account Preference For Zoom B (Preference = 4)

```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
You have chosen the Search By Zoom Account Options
Now key in the zoom account you want to search. Please Use CAPS LOCK (A/B) = B

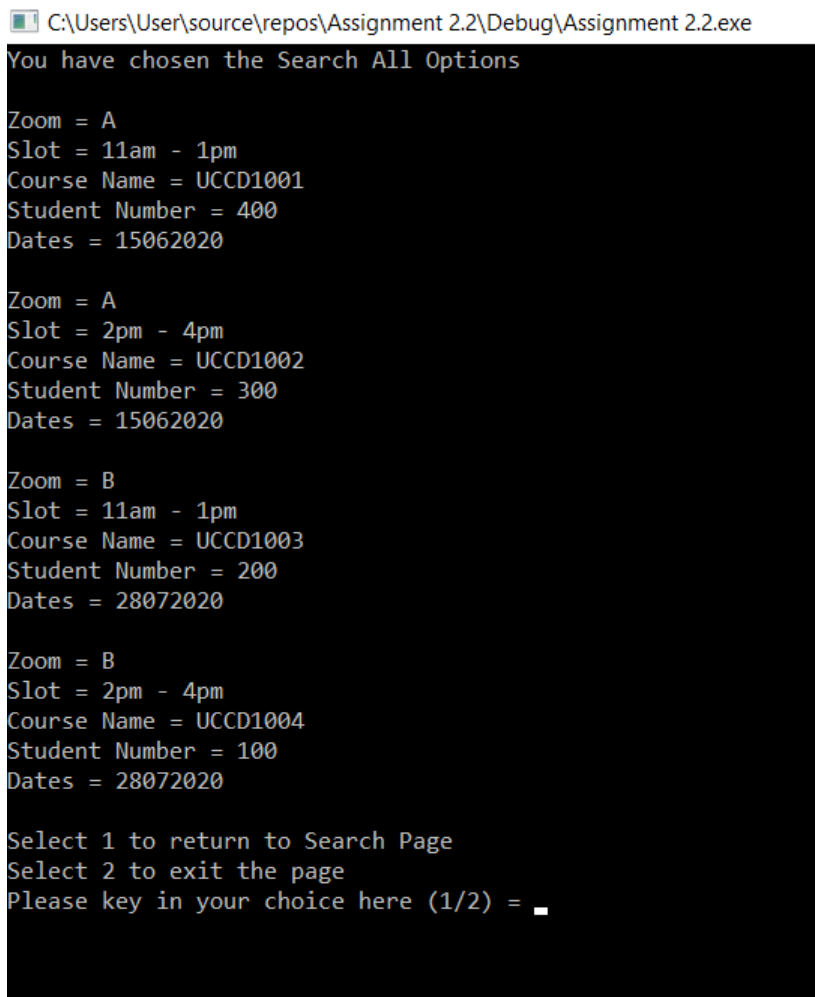
Zoom = B
Slot = 11am - 1pm
Course Name = UCCD1003
Student Number = 200
Dates = 28072020

Zoom = B
Slot = 2pm - 4pm
Course Name = UCCD1004
Student Number = 100
Dates = 28072020

Select 1 to return to Search Page
Select 2 to exit the page
Please key in your choice here (1/2) =
```

Image 6.6 – Zoom Account Preference Interface

G.) The Search All Preference (Preference = 5)



```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
You have chosen the Search All Options

Zoom = A
Slot = 11am - 1pm
Course Name = UCCD1001
Student Number = 400
Dates = 15062020

Zoom = A
Slot = 2pm - 4pm
Course Name = UCCD1002
Student Number = 300
Dates = 15062020

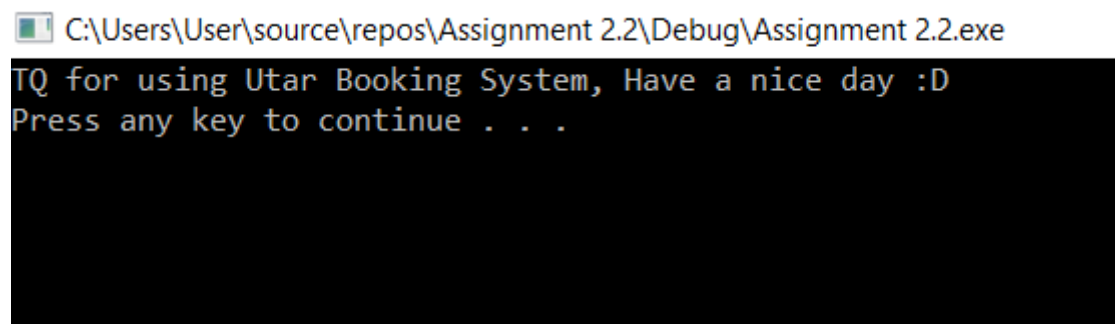
Zoom = B
Slot = 11am - 1pm
Course Name = UCCD1003
Student Number = 200
Dates = 28072020

Zoom = B
Slot = 2pm - 4pm
Course Name = UCCD1004
Student Number = 100
Dates = 28072020

Select 1 to return to Search Page
Select 2 to exit the page
Please key in your choice here (1/2) = 
```

Image 6.7 – The Search All Preference Interface

H.) The Exit Preference (Preference = 6)



```
C:\Users\User\source\repos\Assignment 2.2\Debug\Assignment 2.2.exe
TQ for using Utar Booking System, Have a nice day :D
Press any key to continue . . .
```

Image 6.8 – The Exit Preference Interface

Part E- Pseudocode

Pseudocode

Declare variables and functions with suitable data types.

Main Function

Declare variables with suitable data types.

INPUTS course code and name, number of students, and date

IF number of students is more than 250 and less than 501

Function name: category_A

Pass in: course_name: string, student_num: integer, date: string, start: integer

OUTPUT the available slots for category A and ask user to input the time

IF time is 11a.m. - 1p.m. and zoom is A

RUN function slot1

IF time is 2p.m. - 4p.m. and zoom is A

RUN function slot2

END Function category_A

Pass out: course_name: string, student_num: integer, date: string, start: integer

ELSE IF number of students is more than 0 and less than or equal to 250

Function name: category_B

Pass in: course_name: string, student_num: integer, date: string, start: integer

OUTPUT the available slots for category A and ask user to input the time

IF time is 11a.m. - 1p.m. and zoom is A

RUN function slot1

IF time is 2p.m. - 4p.m. and zoom is A

RUN function slot2

IF time is 11a.m. - 1p.m. and zoom is B

RUN function slot3

IF time is 2p.m. - 4p.m. and zoom is B

RUN function slot4

END Function category_B

Pass out: course_name: string, student_num: integer,
date: string, start: integer

RUN function output

Function name: slot1

Declare variables with suitable data types.

Pass in: start: integer, booking_1: string, zoom: string, slot:
string, course_name: string, student_num: integer, date: string

IF the slot has not been occupied

store the course code and name, number of students, date, zoom
and the time

RUN function record

OUTPUT the course code and name, number of students, date,
zoom and the time and ask the user either continue booking
exit the program

IF choose to continue booking

REPEAT main function

ELSE exit the program

END main function

IF the slot has been occupied

DISPLAY the course code and name that occupy the slot and ask
to remain, update or cancel

IF remain

remain the previous booking that have been made earlier

OUTPUT the course code and name, number of students,
date, zoom and the time that have been booked previously
and ask the user either continue booking exit the program

IF update

RUN function update

OUTPUT the course code and name, number of students,
date, zoom and the time that have been updated and ask
the user either continue booking exit the program

IF cancel


```

        RUN function cancel

OUTPUT the course code and name, number of students, date,
zoom and the time and ask the user either continue booking
exit the program

IF choose to continue booking

        REPEAT main function

ELSE exit the program

        END main function

END Function slot1

Pass out: start: integer, booking_1: string


Function name: slot2

Declare variables with suitable data types.

Pass in: start: integer, booking_2: string, zoom: string, slot:
string, course_name: string, student_num: integer, date: string

IF the slot has not been occupied

    store the course code and name, number of students, date, zoom
    and the time

    RUN function record

    OUTPUT the course code and name, number of students, date,
    zoom and the time and ask the user either continue booking
    exit the program

    IF choose to continue booking

        REPEAT main function

    ELSE exit the program

        END main function

IF the slot has been occupied

    DISPLAY the course code and name that occupy the slot and ask
    to remain, update or cancel

    IF remain

        remain the previous booking that have been made earlier

        OUTPUT the course code and name, number of students,
        date, zoom and the time that have been booked previously
        and ask the user either continue booking exit the program

    IF update

```

```

        RUN function update

        OUTPUT the course code and name, number of students,
        date, zoom and the time that have been updated and ask
        the user either continue booking exit the program

IF cancel

        RUN function cancel

        OUTPUT the course code and name, number of students, date,
        zoom and the time and ask the user either continue booking
        exit the program

IF choose to continue booking

        REPEAT main function

ELSE exit the program

        END main function

END Function slot2

Pass out: start: integer, booking_2: string


Function name: slot3

Declare variables with suitable data types.

Pass in: start: integer, booking_3: string, zoom: string, slot:
string, course_name: string, student_num: integer, date: string

IF the slot has not been occupied

    store the course code and name, number of students, date, zoom
    and the time

    RUN function record

    OUTPUT the course code and name, number of students, date,
    zoom and the time and ask the user either continue booking
    exit the program

    IF choose to continue booking

        REPEAT main function

    ELSE exit the program

        END main function

IF the slot has been occupied

    DISPLAY the course code and name that occupy the slot and ask
    to remain, update or cancel

    IF remain

```

remain the previous booking that have been made earlier

OUTPUT the course code and name, number of students, date, zoom and the time that have been booked previously and ask the user either continue booking exit the program

IF update

RUN function update

OUTPUT the course code and name, number of students, date, zoom and the time that have been updated and ask the user either continue booking exit the program

IF cancel

RUN function cancel

OUTPUT the course code and name, number of students, date, zoom and the time and ask the user either continue booking exit the program

IF choose to continue booking

REPEAT main function

ELSE exit the program

END main function

END Function slot3

Pass out: start: integer, booking_3: string

Function name: slot4

Declare variables with suitable data types.

Pass in: start: integer, booking_4: string, zoom: string, slot: string, course_name: string, student_num: integer, date: string

IF the slot has not been occupied

store the course code and name, number of students, date, zoom and the time

RUN function record

OUTPUT the course code and name, number of students, date, zoom and the time and ask the user either continue booking exit the program

IF choose to continue booking

REPEAT main function

ELSE exit the program

```

        END main function

IF the slot has been occupied

    DISPLAY the course code and name that occupy the slot and ask
    to remain, update or cancel

    IF remain

        remain the previous booking that have been made earlier

        OUTPUT the course code and name, number of students,
        date, zoom and the time that have been booked previously
        and ask the user either continue booking exit the program

    IF update

        RUN function update

        OUTPUT the course code and name, number of students,
        date, zoom and the time that have been updated and ask
        the user either continue booking exit the program

    IF cancel

        RUN function cancel

        OUTPUT the course code and name, number of students, date,
        zoom and the time and ask the user either continue booking
        exit the program

    IF choose to continue booking

        REPEAT main function

    ELSE exit the program

        END main function

END Function slot4

Pass out: start: integer, booking_4: string

```

Function name: output

Declare variables with suitable data types.

Pass in: Nothing

Open booking.txt file and **INPUT** all the booking details into the program

DISPLAY the instructions and ask the user to search for the all the booking details or search by date, course code and name, time, or zoom account or exit the program

IF search by date

```
    Ask to INPUT the date the user wants to search for
    DISPLAY all the bookings details with the selected date
    Ask the user if he/she wants to continue booking or exit the
search system

    IF continue

        REPEAT function output

    ELSE IF exit

        END function output
IF search by course code and name
    Ask to INPUT the course code and name the user wants to search
for
    DISPLAY all the bookings details with the selected course code
and name
    Ask the user if he/she wants to continue booking or exit the
search system

    IF continue

        REPEAT function output

    ELSE IF exit

        END function output
IF search by time
    Ask to INPUT the time period the user wants to search for
    DISPLAY all the bookings details with the selected time period
    Ask the user if he/she wants to continue booking or exit the
search system

    IF continue

        REPEAT function output

    ELSE IF exit

        END function output
IF search by zoom account
    Ask to INPUT the zoom account the user wants to search for
    DISPLAY all the booking details with the selected zoom account
```

Ask the user if he/she wants to continue booking or exit the search system

IF continue

REPEAT function output

ELSE IF exit

END function output

IF search for all

DISPLAY all the booking details in the booking.txt file

Ask the user if he/she wants to continue booking or exit the search system

IF continue

REPEAT function output

ELSE IF exit

END function output

IF exit the program

END function output

Pass out: Nothing

Function name: record

Declare variables with suitable data types.

Pass in: zoom: string, slot: string, course_name: string,
student_num: integer, date: string

IF the booking.txt file has not been created

Create the file and store all the booking data into the booking.txt file

ELSE IF the booking.txt file has been created

Store the booking data into the booking.txt file follow by the next line

END Function record

Pass out: Nothing

Function name: update

Declare variables with suitable data types.

Pass in: course_name: string, student_num: integer, zoom: string,
slot: string, date: string

Open booking.txt file and **INPUT** all the booking details into the
program

REPEAT

From the booking details, search for the set of booking that
do not match the date or the zoom account or the time period
of the current booking

IF the set of booking is the first set of booking that do not
match the date or the zoom account or the time period of the
current booking

Open a new booking.txt file and store the set of booking
data into the booking.txt file

ELSE

store the set of booking data into the existing
booking.txt file

RUN function record

store the course code and name, number of students, date, zoom and
the time

END function update

Pass out: course_name: string, student_num: integer, zoom: string,
slot: string, date: string

Function name: cancel

Declare variables with suitable data types.

Pass in: course_name: string, student_num: integer, zoom: string,
slot: string, date: string

Open booking.txt file and **INPUT** all the booking details into the
program

REPEAT

From the booking details, search for the set of booking that
do not match the date or the zoom account or the time period
of the current booking

IF the set of booking is the first set of booking that do not match the date or the zoom account or the time period of the current booking

Open a new booking.txt file and store the set of booking data into the booking.txt file

ELSE

store the set of booking data into the existing booking.txt file

END function cancel

Pass out: course_name: string, student_num: integer, zoom: string, slot: string, date: string

Part F – Source Code

```
#include <iostream>
#include <iomanip>
#include <fstream>
#include <string>
#include <cctype>
using namespace std;

void category_A(string&, int&, string&, int&);
void category_B(string&, int&, string&, int&);
void record(string, string, string, int, string);
void slot1(int&, string&, string, string, string, int, string);
void slot2(int&, string&, string, string, string, int, string);
void slot3(int&, string&, string, string, string, int, string);
void slot4(int&, string&, string, string, string, int, string);
void update(string&, int&, string&, string&, string&);
void cancel(string&, int&, string&, string&, string&);
void output();
struct Booking_num
{
    string zoom;
    string slot;
    string course_name;
    string student_num;
    string date;
};
int verification_date(string);

string booking_1;
string booking_2;
string booking_3;
string booking_4;
string date_slot[5][100] = { {"15062020", "16062020",
"17062020", "18062020", "19062020", "22062020", "23062020", "24062020",
"25062020", "26062020", "29062020",
"30062020", "01072020", "02072020", "03072020", "06072020", "07072020", "08072020",
", "09072020", "10072020",
"13072020", "14072020", "15072020", "16072020", "17072020", "20072020", "21072020",
", "22072020", "23072020", "24072020", "27072020", "28072020", "29072020", "30072020",
"03082020", "04082020", "05082020", "06082020", "07082020", "10082020", "11082020",
"12082020", "13082020", "14082020", "17082020", "18082020", "19082020", "20082020",
"21082020", "24082020", "25082020", "26082020", "27082020", "28082020", "31082020",
"01092020", "02092020", "03092020", "04092020", "07092020", "08092020", "09092020",
"10092020", "11092020", "14092020", "15092020", "16092020", "17092020", "18092020" }, {}, {}, {}, {}, {} };
string book_name[4][100];
int book_num[4][100];
Booking_num booking[100];
```

```

int ruc;

int main()
{
    int start = 1;
    string course_name;
    int student_num;
    string date;
    int confirm;

    while (start != 2)
    {
        cout << "\t\t~~~~~ =====
~~~~~" << endl;
        cout << "\t\t~~~~~ |Welcome to UTAR Booking Center|
~~~~~" << endl;
        cout << "\t\t~~~~~ =====
~~~~~" << "\n\n";
        cout << "Due to recent pandemic outbreak , UTAR had decided
to carry out online classes for June trimester starting from 15 June 2020
to 20 September 2020." << endl;
        cout << "Lecturer should book zoom accounts to carry on
their teaching through online classes." << "\n\n";
        cout << "Maximum Student per class is 500" << endl << endl;
        cout << "Please enter the name of courses = ";
        cin.clear();
        getline(cin, course_name);
        cout << endl;
        cout << "Please enter the number of students = ";
        cin >> student_num;
        if (student_num <= 0 or student_num > 501)
        {
            cout << "The number of student you key in is
invalid, Please key in between 1-500. TQ :D" << endl;
            system("pause");
            cin.ignore();
            system("cls");
            continue;
        }
        cout << endl;
        cout << "Please enter the date for your booking (DDMMYYYY)
eg. (28072020) = ";
        cin.ignore();
        cin >> date;
        confirm = verification_date(date);
        if (confirm != 1)
        {
            cout << "The date you enter currently is not
available, Please enter a new date again" << endl;

```

```

        start = 1;
        system("pause");
        cin.ignore();
        system("cls");
        continue;
    }
    if (student_num > 250 and student_num < 501)
    {
        system("cls");
        category_A(course_name, student_num, date, start);
    }
    else if (student_num > 0 and student_num <= 250)
    {
        system("cls");
        category_B(course_name, student_num, date, start);
    }
    system("cls");
}
output();

system("pause");
return 0;
}

void category_A(string& course_name, int& student_num, string& date, int&
start)
{
    string zoom;
    string time;

    cout << "The Available Slots For This Course Are" << "\n\n";
    cout << "===== " << endl;
    cout << "Zoom Accounts          Time          " << endl;
    cout << "1.) Zoom A          11am - 1pm          " << endl;
    cout << "2.) Zoom A          2pm - 4pm          " << endl;
    cout << "===== " << endl;
    cout << "Please Select The Available Slot (Please State the Time)"
<< endl;
    zoom = 'A';
    cout << endl;
    cin.ignore();
    cout << "Please Key in Time Available (with am/pm , exactly as
stated above) = ";
    getline(cin, time);
    if (time == "11am - 1pm" && zoom == "A") // slot 1
    {
        string slot = "11am - 1pm";
        slot1(start, booking_1, zoom, slot, course_name,
student_num, date);
    }
}

```

```

    }

    else if (time == "2pm - 4pm" && zoom == "A") // slot 2
    {
        string slot = "2pm - 4pm";
        slot2(start, booking_2, zoom, slot, course_name,
student_num, date);
    }
    else
    {
        cout << "The time you key in is unavailable, please try
again " << endl;
    }
    cin.ignore();
}

void category_B(string& course_name, int& student_num, string& date, int&
start)
{
    string zoom;
    string time;

    cout << "The Available Slots For This Course Are" << "\n\n";
    cout << "======" << endl;
    cout << "Zoom Accounts          Time" << endl;
    cout << "1.) Zoom A          11am - 1pm" << endl;
    cout << "2.) Zoom A          2pm - 4pm" << endl;
    cout << "3.) Zoom B          11am - 1pm" << endl;
    cout << "4.) Zoom B          2pm - 4pm" << endl;
    cout << "======" << endl;
    cout << "Please Select The Available Slot (Please State the Time)"
<< endl;
    cout << "Please key in the Zoom Account (A/B) = ";
    cin >> zoom;
    if (zoom != "A" || zoom != "B")
    cout << endl;
    cin.ignore();
    cout << "Please Key in Time Available (with am/pm , exactly as
stated above) = ";
    getline(cin, time);
    if (time == "11am - 1pm" && zoom == "A") // slot 1
    {
        string slot = "11am - 1pm";
        slot1(start, booking_1, zoom, slot, course_name,
student_num, date);
    }

    else if (time == "2pm - 4pm" && zoom == "A") // slot 2
    {
        string slot = "2pm - 4pm";

```

```

        slot2(start, booking_2, zoom, slot, course_name,
student_num, date);
    }
    else if (time == "11am - 1pm" && zoom == "B") // slot 3
    {
        string slot = "11am - 1pm";
        slot3(start, booking_3, zoom, slot, course_name,
student_num, date);
    }
    else if (time == "2pm - 4pm" && zoom == "B") // slot 4
    {
        string slot = "2pm - 4pm";
        slot4(start, booking_4, zoom, slot, course_name,
student_num, date);
    }
    else
    {
        cout << "The time you key in or the zoom account is
unavailable, please try again " << endl;
    }
    cin.ignore();
}

void slot1(int& start, string& booking_1, string zoom, string slot, string
course_name, int student_num, string date)
{
    int date_space = 1;
    for (int j = 0; j < 100; j++)
    {
        if (date == date_slot[0][j])
        {
            if (date_slot[1][j] != "1")
            {
                date_slot[1][j] = "1";
                book_name[0][j] = course_name;
                book_num[0][j] = student_num;
                cout << "Booking Completed and Recorded";
                record(zoom, slot, course_name, student_num,
date);

                cout << endl;
                cout << "\n Thank you for booking!" << endl;
                cout << "----- "

<< endl;

                cout << " Your Booking are listed below:
\n\n" << " Course code and name: " << course_name << endl;
                cout << " Number of Students: " <<
student_num << endl;

```

```

        cout << " Zoom Class: " << zoom << '\n' << "
Time: " << slot << '\n' << " Date: " << date << '\n' << endl << endl <<
endl;

        cout << "Do you now wish to continue to book
or exit the program?" << endl;
        cout << "Press 1 if you wish to continue the
booking system " << endl;
        cout << "Press 2 if you proceed to search
part" << endl;

        cout << "Your pick is = ";
        cin >> start;
        booking_1 = course_name;
    }
    else
    {
        if (booking_1 == course_name)
        {
            cout << "
===== " << endl;
            cout << " You have already book this
zoom slot previously " << endl;
            cout << "
===== " << endl;
        }
        else
        {
            cout << "Unfortunately this current
slot is occupied by " << book_name[0][j] << endl;
            cout << "Press 1 to Remain (There
will be no changes, everything remain unchange)" << endl;
            cout << "Press 2 to Update (Replace
the current data with the previous data)" << endl;
            cout << "Press 3 to Cancel (Remove
the current data and previous data)" << endl;
            cout << "Your Selection Is = ";
            cin >> ruc;
            if (ruc == 1)
            {
                cout << "\n Your previous
booking is remained." << endl;
                cout << "-----
----- " << endl;
                cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << book_name[0][j] <<
endl;
                cout << " Number of Students:
" << book_num[0][j] << endl;
            }
        }
    }
}

```

```

                                cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

                                }
                                if (ruc == 2)
                                {
                                    update(course_name,
student_num, zoom, slot, date);

                                    cout << "\n Your booking has
been updated " << endl;

                                    cout << "-----
----- " << endl;

                                    cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << course_name << endl;
                                    cout << " Number of Students:
" << student_num << endl;

                                    cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

                                }
                                if (ruc == 3)
                                {
                                    date_slot[1][j] = "0";
                                    cancel(course_name,
student_num, zoom, slot, date);

                                    cout << "\n Your bookings has
been cancelled" << endl;

                                    cout << "-----
----- " << endl;

                                    cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << "-" << endl;
                                    cout << " Number of Students:
" << "-" << endl;

                                    cout << " Zoom Class: " << "-"
" << '\n' << " Time: " << "-" << '\n' << " Date: " << "-" << '\n' << endl;
                                }
                                else if (!(ruc == 1 || ruc == 2 ||
ruc == 3))
                                {
                                    cout << "The selection you
key in is unavailable, Please try again." << endl;
                                    system("pause");
                                    continue;
                                }
                                cout << endl;
                                cout << "Do you now wish to continue
to book or exit the program?" << endl;
                                cout << "Press 1 if you wish to
continue the booking system " << endl;

```

```

        cout << "Press 2 if you proceed to
search part" << endl;

        cout << "Your pick is = ";
        cin >> start;

    }

}

}

}

void slot2(int& start, string& booking_2, string zoom, string slot, string
course_name, int student_num, string date)
{
    int date_space = 2;
    for (int j = 0; j < 100; j++)
    {
        if (date == date_slot[0][j])
        {
            if (date_slot[2][j] != "1")
            {
                date_slot[2][j] = "1";
                book_name[1][j] = course_name;
                book_num[1][j] = student_num;
                cout << "Booking Completed and Recorded";
                record(zoom, slot, course_name, student_num,
date);

                cout << endl;
                cout << "\n Thank you for booking!" << endl;
                cout << "----- "

<< endl;

                cout << " Your Booking are listed below:
\n\n" << " Course code and name: " << course_name << endl;
                cout << " Number of Students: " <<
student_num << endl;

                cout << " Zoom Class: " << zoom << '\n' << "
Time: " << slot << '\n' << " Date: " << date << '\n' << endl << endl <<
endl;

                cout << "Do you now wish to continue to book
or exit the program?" << endl;

                cout << "Press 1 if you wish to continue the
booking system " << endl;

                cout << "Press 2 if you proceed to search
part" << endl;

                cout << "Your pick is = ";
                cin >> start;
                booking_2 = course_name;

            }
            else
            {
                if (booking_2 == course_name)

```



```

        {
            cout << "
===== " << endl;
            cout << " You have already book this
zoom slot previously " << endl;
            cout << "
===== " << endl;
        }
        else
        {
            cout << "Unfortunately this current
slot is occupied by " << book_name[1][j] << endl;
            cout << "Press 1 to Remain (There
will be no changes, everything remain unchange)" << endl;
            cout << "Press 2 to Update (Replace
the current data with the previous data)" << endl;
            cout << "Press 3 to Cancel (Remove
the current data and previous data)" << endl;
            cout << "Your Selection Is = ";
            cin >> ruc;
            if (ruc == 1)
            {
                cout << "\n Your previous
booking is remained." << endl;
                cout << "-----
-----" << endl;
                cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << book_name[1][j] <<
endl;
                cout << " Number of Students:
" << book_num[1][j] << endl;
                cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;
            }
            if (ruc == 2)
            {
                update(course_name,
student_num, zoom, slot, date);
                cout << "\n Your booking has
been updated \n" << endl;
                cout << "-----
-----" << endl;
                cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << course_name << endl;
                cout << " Number of Students:
" << student_num << endl;
            }
        }
    }
}

```

```

        cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

    }
    if (ruc == 3)
    {
        date_slot[2][j] = "0";
        cancel(course_name,
student_num, zoom, slot, date);

        cout << "\n Your bookings has
been cancelled" << endl;

        cout << "-----"
-----" << endl;

        cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << "-" << endl;
        cout << " Number of Students:
" << "-" << endl;

        cout << " Zoom Class: " << "-"
" << '\n' << " Time: " << "-" << '\n' << " Date: " << "-" << '\n' << endl;

    }
    else if (!(ruc == 1 || ruc == 2 ||
ruc == 3))
    {
        cout << "The selection you
key in is unavailable, Please try again." << endl;
        system("pause");
        continue;
    }
    cout << endl;
    cout << "Do you now wish to continue
to book or exit the program?" << endl;
    cout << "Press 1 if you wish to
continue the booking system " << endl;
    cout << "Press 2 if you proceed to
search part" << endl;

    cout << "Your pick is = ";
    cin >> start;
}
}
}

}

void slot3(int& start, string& booking_3, string zoom, string slot, string
course_name, int student_num, string date)
{
    int date_space = 3;
    for (int j = 0; j < 100; j++)

```

```

{
    if (date == date_slot[0][j])
    {
        if (date_slot[3][j] != "1")
        {
            date_slot[3][j] = "1";
            book_name[2][j] = course_name;
            book_num[2][j] = student_num;
            cout << "Booking Completed and Recorded";
            record(zoom, slot, course_name, student_num,
date);

            cout << endl;
            cout << "\n Thank you for booking! " <<

endl;

            cout << "----- "

<< endl;

            cout << " Your Booking are listed below:
\n\n" << " Course code and name: " << course_name << endl;
            cout << " Number of Students: " <<
student_num << endl;

            cout << " Zoom Class: " << zoom << '\n' << "
Time: " << slot << '\n' << " Date: " << date << '\n' << endl << endl <<
endl;

            cout << "Do you now wish to continue to book
or exit the program?" << endl;
            cout << "Press 1 if you wish to continue the
booking system " << endl;

            cout << "Press 2 if you proceed to search
part" << endl;

            cout << "Your pick is = ";
            cin >> start;
            booking_3 = course_name;
        }
        else
        {
            if (booking_3 == course_name)
            {
                cout << "
===== " << endl;
                cout << " You have already book this
zoom slot previously " << endl;

                cout << "
===== " << endl;
            }
            else
            {
                cout << "Unfortunately this current
slot is occupied by " << book_name[2][j] << endl;

```

```

        cout << "Press 1 to Remain (There
will be no changes, everything remain unchange)" << endl;
        cout << "Press 2 to Update (Replace
the current data with the previous data)" << endl;
        cout << "Press 3 to Cancel (Remove
the current data and previous data)" << endl;
        cout << "Your Selection Is = ";
        cin >> ruc;
        if (ruc == 1)
        {
            cout << "\n Your previous
booking is remained." << endl;

            cout << "-----
----- " << endl;

            cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << book_name[2][j] <<
endl;

            cout << " Number of Students:
" << book_num[2][j] << endl;

            cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

        }
        if (ruc == 2)
        {
            update(course_name,
student_num, zoom, slot, date);

            cout << "\n Your booking has
been updated" << endl;

            cout << "-----
----- " << endl;

            cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << course_name << endl;
            cout << " Number of Students:
" << student_num << endl;

            cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

        }
        if (ruc == 3)
        {
            date_slot[3][j] = "0";
            cancel(course_name,
student_num, zoom, slot, date);

            cout << "\n Your bookings has
been cancelled" << endl;

```

```

cout << "-----"
-----" << endl;

cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << "-" << endl;
cout << " Number of Students:
" << "-" << endl;

cout << " Zoom Class: " << "-"
" << '\n' << " Time: " << "-" << '\n' << " Date: " << "-" << '\n' << endl;

}
else if (!(ruc == 1 || ruc == 2 ||
ruc == 3))
{
cout << "The selection you
key in is unavailable, Please try again." << endl;
system("pause");
continue;
}
cout << endl;
cout << "Do you now wish to continue
to book or exit the program?" << endl;
cout << "Press 1 if you wish to
continue the booking system " << endl;
cout << "Press 2 if you proceed to
search part" << endl;

cout << "Your pick is = ";
cin >> start;
}
}
}

}

void slot4(int& start, string& booking_4, string zoom, string slot, string
course_name, int student_num, string date)
{
int date_space = 4;
for (int j = 0; j < 100; j++)
{
if (date == date_slot[0][j])
{
if (date_slot[4][j] != "1")
{
date_slot[4][j] = "1";
book_name[3][j] = course_name;
book_num[3][j] = student_num;
cout << "Booking Completed and Recorded";
record(zoom, slot, course_name, student_num,
date);

cout << endl;

```

```

        cout << "\n Thank you for booking!" << endl;
        cout << "----- "
<< endl;

        cout << " Your Booking are listed below:
\n\n" << " Course code and name: " << course_name << endl;
        cout << " Number of Students: " <<
student_num << endl;

        cout << " Zoom Class: " << zoom << '\n' << "
Time: " << slot << '\n' << " Date: " << date << '\n' << endl << endl <<
endl;

        cout << "Do you now wish to continue to book
or exit the program?" << endl;
        cout << "Press 1 if you wish to continue the
booking system " << endl;
        cout << "Press 2 if you proceed to search
part" << endl;

        cout << "Your pick is = ";
        cin >> start;
        booking_4 = course_name;
    }
    else
    {
        if (booking_4 == course_name)
        {
            cout << "
===== " << endl;
            cout << " You have already book this
zoom slot previously " << endl;
            cout << "
===== " << endl;
        }
        else
        {
            cout << "Unfortunately this current
slot is occupied by " << book_name[3][j] << endl;
            cout << "Press 1 to Remain (There
will be no changes, everything remain unchange)" << endl;
            cout << "Press 2 to Update (Replace
the current data with the previous data)" << endl;
            cout << "Press 3 to Cancel (Remove
the current data and previous data)" << endl;
            cout << "Your Selection Is = ";
            cin >> ruc;
            if (ruc == 1)
            {
                cout << "\n Your previous
booking is remained." << endl;
                cout << "-----
----- " << endl;

```

```

                                cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << book_name[3][j] <<
endl;

                                cout << " Number of Students:
" << book_num[3][j] << endl;

                                cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

                                }
                                if (ruc == 2)
                                {
                                    update(course_name,
student_num, zoom, slot, date);

                                    cout << "\n Your booking has
been updated" << endl;

                                    ----- " << endl;

                                    cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << course_name << endl;
                                    cout << " Number of Students:
" << student_num << endl;

                                    cout << " Zoom Class: " <<
zoom << '\n' << " Time: " << slot << '\n' << " Date: " << date << '\n' <<
endl;

                                }
                                if (ruc == 3)
                                {
                                    date_slot[4][j] = "0";
                                    cancel(course_name,
student_num, zoom, slot, date);

                                    cout << "\n Your bookings has
been cancelled" << endl;

                                    ----- " << endl;

                                    cout << " Your Booking are
listed below: \n\n" << " Course code and name: " << "-" << endl;
                                    cout << " Number of Students:
" << "-" << endl;

                                    cout << " Zoom Class: " << "-"
" << '\n' << " Time: " << "-" << '\n' << " Date: " << "-" << '\n' << endl;

                                }
                                else if (!(ruc == 1 || ruc == 2 ||
ruc == 3))
                                {
                                    cout << "The selection you
key in is unavailable, Please try again." << endl;
                                    system("pause");

```

```

                                continue;
                                }
                                cout << endl;
                                cout << "Do you now wish to continue
to book or exit the program?" << endl;
                                cout << "Press 1 if you wish to
continue the booking system " << endl;
                                cout << "Press 2 if you proceed to
search part" << endl;

                                cout << "Your pick is = ";
                                cin >> start;

                                }
                                }
                                }
                                }
}

```

```

void record(string zoom, string slot, string course_name, int student_num,
string date)

```

```

{
    ofstream booking_file;
    booking_file.open("booking.txt", ios::app);
    if (booking_file.is_open())
    {
        booking_file << zoom << endl;
        booking_file << slot << endl;
        booking_file << course_name << endl;
        booking_file << student_num << endl;
        booking_file << date << endl;
    }
    booking_file.close();
}

```

```

int verification_date(string date)

```

```

{
    int confirm = 0;
    for (int j = 0; j < 100; j++)
    {
        if (date == date_slot[0][j])
        {
            confirm = confirm + 1;
        }
    }
    return confirm;
}

```

```

void update(string& course_name, int& student_num, string& zoom, string&
slot, string& date)

```

```

{
    int x = 0;

```



```

int j = 0;
int b = 1;
ifstream booking_file;
booking_file.open("booking.txt");
if (booking_file.is_open())
{
    while (!booking_file.eof())
    {
        getline(booking_file, booking[x].zoom);
        getline(booking_file, booking[x].slot);
        getline(booking_file, booking[x].course_name);
        getline(booking_file, booking[x].student_num);
        getline(booking_file, booking[x].date);
        x++;
    }
    booking_file.close();
}
x = x - 2;
while (j <= x)
{
    if (date == booking[j].date && zoom == booking[j].zoom &&
slot == booking[j].slot && j == 0)
    {
        ofstream booking_file;
        booking_file.open("booking.txt", ios::trunc);
        if (booking_file.is_open())
        {
        }
        booking_file.close();
    }
    else if (date != booking[j].date || zoom != booking[j].zoom
|| slot != booking[j].slot)
    {
        if (b == 1)
        {
            ofstream booking_file;
            booking_file.open("booking.txt",
ios::trunc);

            if (booking_file.is_open())
            {
                booking_file << booking[j].zoom <<
endl;
                booking_file << booking[j].slot <<
endl;
                booking_file <<
booking[j].course_name << endl;
                booking_file <<
booking[j].student_num << endl;

```



```

        book_num[2][j] = student_num;
        booking_3 = course_name;
    }
    if (date == date_slot[0][j] and zoom == "B" and slot ==
"2pm - 4pm")
    {
        book_name[3][j] = course_name;
        book_num[3][j] = student_num;
        booking_4 = course_name;
    }
}
}
void cancel(string& course_name, int& student_num, string& zoom, string&
slot, string& date)
{
    int x = 0;
    int j = 0;
    int c = 1;
    ifstream booking_file;
    booking_file.open("booking.txt");
    if (booking_file.is_open())
    {
        while (!booking_file.eof())
        {
            getline(booking_file, booking[x].zoom);
            getline(booking_file, booking[x].slot);
            getline(booking_file, booking[x].course_name);
            getline(booking_file, booking[x].student_num);
            getline(booking_file, booking[x].date);
            x++;
        }
        booking_file.close();
    }
    x = x - 2;
    while (j <= x)
    {
        if (date == booking[j].date && zoom == booking[j].zoom &&
slot == booking[j].slot && j == 0)
        {
            ofstream booking_file;
            booking_file.open("booking.txt", ios::trunc);
            if (booking_file.is_open())
            {
            }
            booking_file.close();
        }
        else if (date != booking[j].date || zoom != booking[j].zoom
|| slot != booking[j].slot)
        {

```

```

        if (c == 1)
        {
            ofstream booking_file;
            booking_file.open("booking.txt",
ios::trunc);

            if (booking_file.is_open())
            {
                booking_file << booking[j].zoom <<
endl;
                booking_file << booking[j].slot <<
endl;
                booking_file <<
booking[j].course_name << endl;
                booking_file <<
booking[j].student_num << endl;
                booking_file << booking[j].date <<
endl;
            }
            booking_file.close();
            c = c + 1;
        }
        else
        {
            ofstream booking_file;
            booking_file.open("booking.txt", ios::app);
            if (booking_file.is_open())
            {
                booking_file << booking[j].zoom <<
endl;
                booking_file << booking[j].slot <<
endl;
                booking_file <<
booking[j].course_name << endl;
                booking_file <<
booking[j].student_num << endl;
                booking_file << booking[j].date <<
endl;
            }
            booking_file.close();
        }
        j++;
    }
}

void output()
{
    int x = 0;

```

```

int selection;
int choice = 0;
string dates;
string course_name;
string time;
string zoom;

ifstream booking_file;
booking_file.open("booking.txt");
if (booking_file.is_open())
{
    while (!booking_file.eof())
    {
        getline(booking_file, booking[x].zoom);
        getline(booking_file, booking[x].slot);
        getline(booking_file, booking[x].course_name);
        getline(booking_file, booking[x].student_num);
        getline(booking_file, booking[x].date);
        x++;
    }
    booking_file.close();
}

while (choice != 2)
{
    cout << "\t\t *&?!  =====  !?#" << endl;
    cout << "\t\t  #@  |Search Engine|  @!%" << endl;
    cout << "\t\t  %&&  =====  *$#" << endl;
    cout << "There are a few preferences for searching to ease
your findings for bookings :D" << endl << endl;
    cout << "\t\t||SEARCH PREFERENCES||" << endl;
    cout << "1.) Search By Dates" << endl;
    cout << "2.) Search By Course Name" << endl;
    cout << "3.) Search By Time Range" << endl;
    cout << "4.) Search By Zoom Account" << endl;
    cout << "5.) Search All (Every Booking)" << endl;
    cout << "6.) Exit Page" << endl;

    cout << "Please key in your preference (1/2/3/4/5/6) = ";
    cin >> selection;
    system("cls");
    if (selection == 1) // date
    {
        cout << "You have chosen the Search By Dates
Options" << endl;

        cout << "Now key in the date you want to search = ";
        cin.ignore();
        getline(cin, dates);
        cout << endl;
        int j = 0;
    }
}

```

```

        while (j <= x)
        {
            if (dates == booking[j].date)
            {
                cout << "Zoom = " << booking[j].zoom
<< endl;
                cout << "Slot = " << booking[j].slot
<< endl;
                cout << "Course Name = " <<
booking[j].course_name << endl;
                cout << "Student Number = " <<
booking[j].student_num << endl;
                cout << "Dates = " << booking[j].date
<< endl << endl;
                j++;
            }
            else
            {
                j++;
            }
        }
        cout << "Select 1 to return to Search Page" << endl;
        cout << "Select 2 to exit the page" << endl;
        cout << "Please key in your choice here (1/2) = ";
        cin >> choice;
        system("cls");
    }
    else if (selection == 2) //course_name
    {
        cout << "You have chosen the Search By Course Name
Options" << endl;
        cout << "Now key in the course name you want to
search = ";
        cin.ignore();
        getline(cin, course_name);
        cout << endl;
        int j = 0;
        while (j <= x)
        {
            if (course_name == booking[j].course_name)
            {
                cout << "Zoom = " << booking[j].zoom
<< endl;
                cout << "Slot = " << booking[j].slot
<< endl;
                cout << "Course Name = " <<
booking[j].course_name << endl;

```

```

        cout << "Student Number = " <<
booking[j].student_num << endl;
        cout << "Dates = " << booking[j].date
<< endl << endl;

        j++;
    }
    else
    {
        j++;
    }
}
cout << "Select 1 to return to Search Page" << endl;
cout << "Select 2 to exit the page" << endl;
cout << "Please key in your choice here (1/2) = ";
cin >> choice;
system("cls");
}
else if (selection == 3) // time
{
    cout << "You have chosen the Search By Time Options"
<< endl;

    cout << "Now key in the time range you want to
search = ";

    cin.ignore();
    getline(cin, time);
    cout << endl;
    int j = 0;
    while (j <= x)
    {
        if (time == booking[j].slot)
        {
            cout << "Zoom = " << booking[j].zoom

            cout << "Slot = " << booking[j].slot

            cout << "Course Name = " <<
booking[j].course_name << endl;
            cout << "Student Number = " <<
booking[j].student_num << endl;
            cout << "Dates = " << booking[j].date
<< endl << endl;

            j++;
        }
        else
        {
            j++;
        }
    }
    cout << "Select 1 to return to Search Page" << endl;

```

```

        cout << "Select 2 to exit the page" << endl;
        cout << "Please key in your choice here (1/2) = ";
        cin >> choice;
        system("cls");
    }
    else if (selection == 4) // zoom account
    {
        cout << "You have chosen the Search By Zoom Account
Options" << endl;
        cout << "Now key in the zoom account you want to
search. Please Use CAPS LOCK (A/B) = ";
        cin.ignore();
        getline(cin, zoom);
        cout << endl;
        int j = 0;
        while (j <= x)
        {
            if (zoom == booking[j].zoom)
            {
                cout << "Zoom = " << booking[j].zoom
<< endl;
                cout << "Slot = " << booking[j].slot
<< endl;
                cout << "Course Name = " <<
booking[j].course_name << endl;
                cout << "Student Number = " <<
booking[j].student_num << endl;
                cout << "Dates = " << booking[j].date
<< endl << endl;
                j++;
            }
            else
            {
                j++;
            }
        }
        cout << "Select 1 to return to Search Page" << endl;
        cout << "Select 2 to exit the page" << endl;
        cout << "Please key in your choice here (1/2) = ";
        cin >> choice;
        system("cls");
    }
    else if (selection == 5) // all
    {
        cout << "You have chosen the Search All Options" <<
endl;

        cout << endl;
        int j = 0;
        while (j < x - 1)

```



```

        {
            cout << "Zoom = " << booking[j].zoom <<
endl;
            cout << "Slot = " << booking[j].slot <<
endl;
            cout << "Course Name = " <<
booking[j].course_name << endl;
            cout << "Student Number = " <<
booking[j].student_num << endl;
            cout << "Dates = " << booking[j].date <<
endl << endl;

            j++;
        }
        cout << "Select 1 to return to Search Page" << endl;
        cout << "Select 2 to exit the page" << endl;
        cout << "Please key in your choice here (1/2) = ";
        cin >> choice;
        system("cls");
    }
    else if (selection == 6) //exit
    {
        choice = 2;
    }
    else
    {
        cout << "The input is invalid, Please try again :D "
<< endl;

        choice = 1;
        system("pause");
        system("cls");
    }
}
cout << "TQ for using Utar Booking System, Have a nice day :D" <<
endl;
}

```