Hong Kong Diploma of Secondary Education

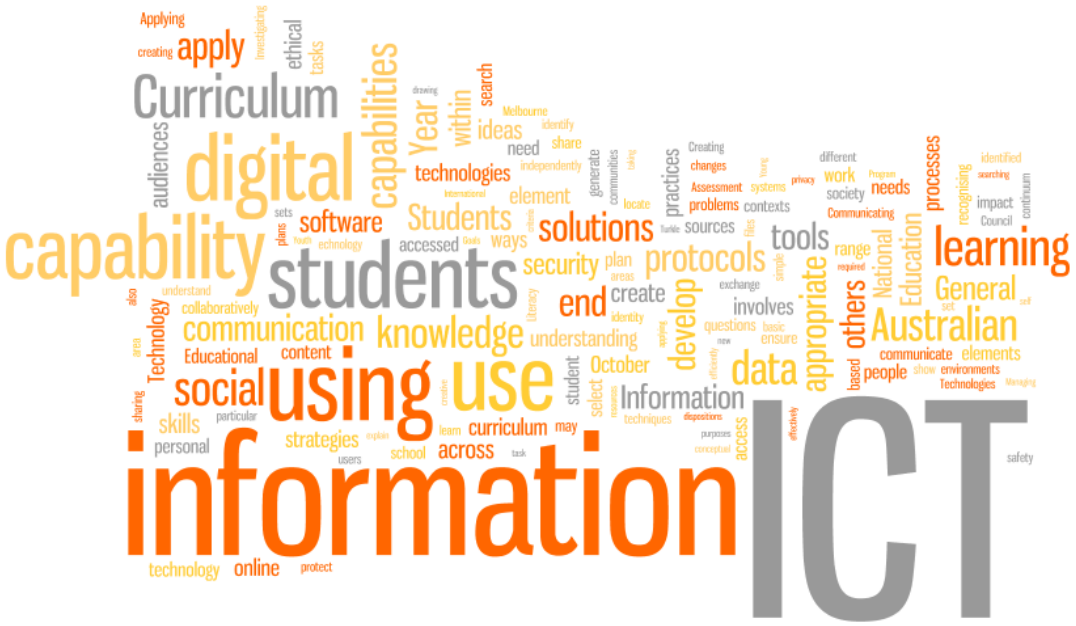
Examination 201x

Information and Communication Technology

(Coursework)

Option D: Software Development

Title: Puzzle Game



Content

Design

The reason that I design a puzzle game is because of my childhood dream. I hope to let the students to have a creative mind and cricital mind in order to duel with the difficult task in the game. competition between students. So, by designing an interesting educational software for students, there is no doubt that it will help students learn better. So, there are different functions in my program, for example, students can login in their own accounts or registering a new account. I also provide a free play for the player that do not need to login in the games. Other functions like displaying their points, change the passcode and logout their account after playing. The programe will save the students work after logout the program.

The program will have following function:

1) Personal accounts for students;

2) Login function;

3) Registering function;

4) Guess number

5) Marks Section

I will assume that:

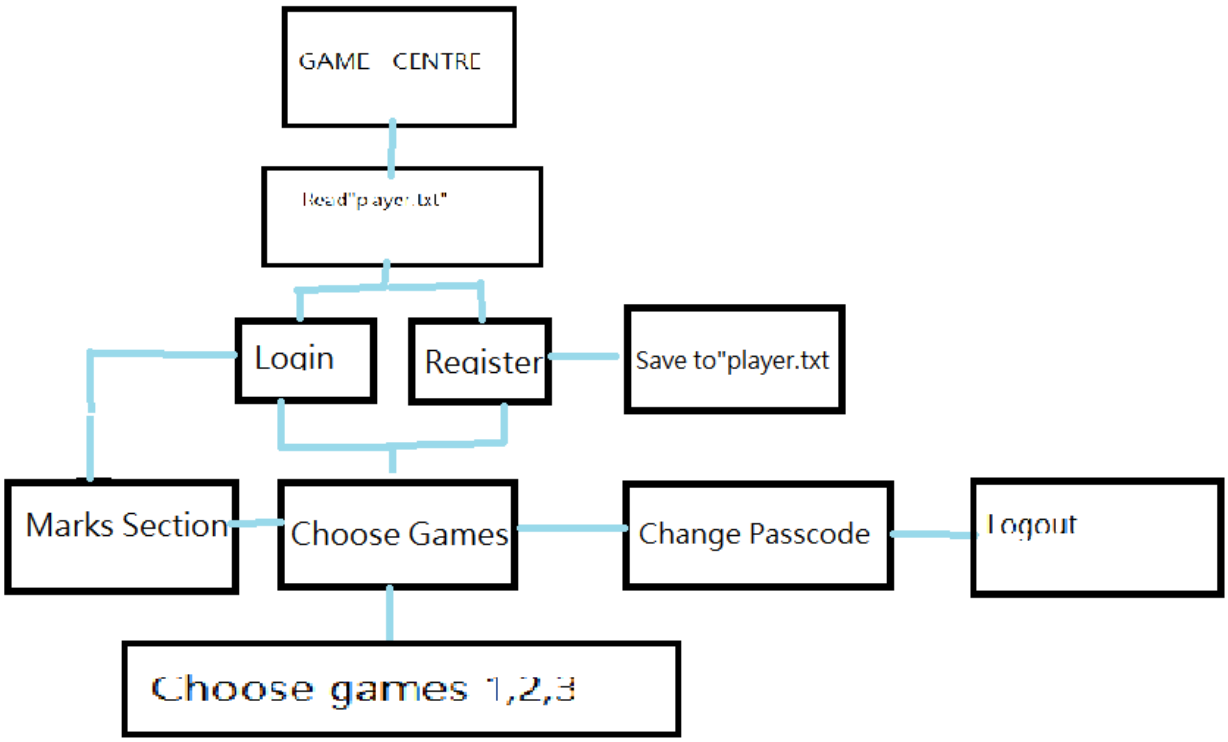
1) Students who

2) are using the software should be F4-F6 studnets and stuying ICT as well

3) There are total 100 studnets who choose ICT

4) All the data will store in C:\

5) All games are in MC type


GAME CENTRE
Read"player.txt"
Login
Register
Save to"player.txt
Marks Section
Choose Games
Change Passcode
Logout
Choose games 1,2,3

Special of the program

The program will be built by command line interface using Dev-Pascal. It provide a chance to reduce the size of program and for easy to editing.

Data flow:

The program will read 1 text files in the same folder

1) Player.txt----storing students's ID, passwords, names

This program will display the corresponding content under commands of the users

The program will make changes in order to save to the files after sutdents and new regsitser have finished and logout the system. Vistor record will not be saved as well.

Data file Formats

Player.txt:

1) This file store the Player ID, the passcode,names
2) This file is to indicate whereter the player has input the correct information to login
3) This file will also saves the new register of the games

Data structure

In this program, I use the file to store the data of different playes names and ID . Also, in the program, I used three array to store the above three data. It is because I think that using the file to store the data can be easily to maintain. As the data in the file can be edit and delete easily. So, I tend to use the file to store the data instead of the record. It is using the file can let the program become simple and more easily to debugging and data validation. So, the using the file format to store the data can be easily to control by the program beginner. So, I

Implementation

In the following section, I will introduce the implementation of my program GameCentre. The following parallel array will be used to store:

playerid : array[1..max_player] of string[4];

playerpw : array[1..max_player] of string[4];

name : array[1.. max_player] of string [25];

sex : array[1..max_player] of char;

playcount : array[1.. max_player] of integer;

score : array[1..max_player] of integer;

player_index, num_player,choice : integer;

Procedures

I will show the program in step by step to show the whole set of software.

1)Read Player

This procedure read “students.txt”

```
procedure read_players_info;
var
  i : integer;
  f : text;
begin
  assign(f, 'player.txt');
  reset(f);
  i := 0;
  while not eof(f) do
    begin
      i := i + 1;
      readln(f, playerid[i], playerpw[i], name[i], sex[i], playcount[i], score[i]);
    end;
  num_player := i;
  close(f)
end;
```

2) Store Player info

This area safe the change of passcode, new register and marks of player to the “player.txt”

```
procedure store_players_info;
var  i : integer;
     f : text;
begin
  assign(f, 'playe.txt');
  rewrite(f);
  for i := 1 to num_player do
    begin
      writeln(f, playerid[i], playerpw[i], name[i], sex[i], playcount[i],' ', score[i]);
    end;
  close(f)
end;
```

3)Funtion: GetPw

This function can hide the passcode in order to increase the privacy and safety for the player.

```
function GetPWord : string;
var
  S : string;
  C : Char;
begin
  S := '';
  repeat
    C := ReadKey;
    if (C <> #10) and (C <> #13) and (C <> #8) then
      begin
        S := S + C;
        write('*');
      end
    else if C = #8 then
      begin
        S[0] := Chr(Length(S) - 1);
        GotoXY(WhereX - 1, WhereY);
        write(' ');
        GotoXY(WhereX - 1, WhereY);
      end;
  until (C = #10) or (C = #13);
  GetPWord := S;
  writeLn;
end;
```

4)Display

It help to show the player’s marks in total, playerID, Name, total playcount.

```
procedure display_player_score(player_index : integer);
 begin
  textcolor(Yellow);
  writeln(' <-------------------------------------------> ');
  writeln(' <-------------------------------------------> ');
  writeln;
  writeln(' *UserID: ', playerid[player_index]:5);
  writeln;
  writeln(' *Name : ', name[player_index]:5);
  writeln;
  writeln(' *Total Playcount : ', playcount[player_index]);
  writeln;
  writeln(' *Your total score : ', score[player_index]);
  writeln;
  writeln(' <-------------------------------------------> ');
  writeln('wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww');
  writeln;
  textcolor(Yellow);
  write(' Press <Enter> to return. ');
  readln
end;
```

4)Change Passcode

In this section, player can change the passcode and it will be store in to the “player.txt”.

```
procedure change_password(player_index : integer);
var
  oldpass, newpass1, newpass2 : string;
  pwchanged : boolean;
begin
  pwchanged := false;
  repeat
    clrscr;
    writeln;
    write('        Please enter your old passcode          : ');
    oldpass := GetPword;
    if oldpass <> playerpw[player_index] then
      begin
        writeln;
        writeln('        Wrong old passcode!');
        write('        Press <Enter> to enter again! ');
        readln
      end
    else
      begin
        writeln;
        write('        Please enter your new password (4 char) : ');
        newpass1 := GetPword;
        if length(newpass1) <> 4 then
          begin
            writeln;
            writeln('        The length of password must be 4!');
```

```
            write('        Press <Enter> to retry. ');
            readln
          end
        else
          begin
            writeln;
            write('        Please enter your new password again    : ');
            newpass2 := GetPword;
            if newpass1 <> newpass2 then
              begin
                writeln;
                writeln('        The new passwords do not match!');
                write('        Press <Enter> to retry. ');
                readln
              end
            else
              begin
                playerpw[player_index] := newpass1;
                store_players_info;
                pwchanged := true;
                writeln;
                writeln('        Password changed.');
                write('        Press <Enter> to return. ');
                readln
              end
          end
      end
  until pwchanged
end;
```

5)Registering

This allow the new player or vistor to joing the system and save their record.

```
procedure create_acc;
var
  loginid, playername, password1, password2 : string;
  sex_in : char;
  id_OK, pw_OK : boolean;
  i : integer;
begin
  clrscr;
  writeln;
  textcolor(Yellow);
  writeln('        Your Name with more than 25 characters will be truncated.');
  writeln('        Your Name cannot be changed after input.');
  textcolor(LightGray);
  write('        Please enter your Name (at most 25 char) : ');
  readln(playername);
  repeat
    writeln;
    write('        Please enter your Gender (M or F) : ');
    readln(sex_in)
  until sex_in in ['M', 'F'];

  id_OK := false;
  repeat
    clrscr;
    writeln;
    writeln('        Welcome ', playername, '!');
    writeln;
```

```
  writeln;
  write('        Please enter your PlayerID (4 char) : ');
  readln(loginid);
  if length(loginid) <> 4 then
    begin
      writeln;
      writeln('        The length of PlayerID must be 4!');
      write('        Press <Enter> to retry. ');
      readln
    end
  else
    begin
      id_OK := true;
      for i := 1 to num_player do
        if loginid = playerid[i] then
          begin
            writeln;
            writeln('        The PlayerID has been used!');
            write('        Press <Enter> to retry. ');
            readln;
            id_OK := false
          end
    end
until id_OK;
```

```
 pw_OK := false;
 repeat
   clrscr;
   writeln;
   writeln('        Welcome ', playername, '!');
   writeln;
   writeln('        Your UserID is ', loginid);
   writeln;
   write('        Please enter your password (4 char) : ');
   password1 := GetPword;
   if length(password1) <> 4 then
     begin
       writeln;
       writeln('        The length of password must be 4!');
       write('        Press <Enter> to retry. ');
       readln
     end
       else
         begin
           writeln;
           write('        Please enter your password again    : ');
           password2 := GetPword;
           if password1 <> password2 then
             begin
               writeln;
               writeln('        The passwords do not match!');
               write('        Press <Enter> to retry. ');
```

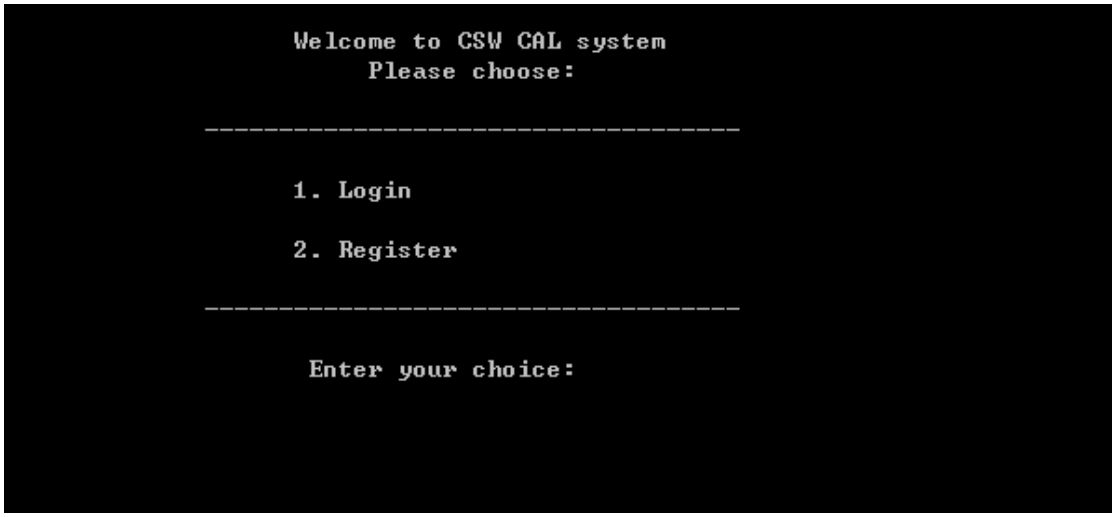
```
            readln
          end
        else
          begin
            num_player := num_player + 1;
            playername := playername + '                          ';
            playername := copy(playername, 1, 25);

            playerid[num_player] := loginid;
            playerpw[num_player] := password1;
            name[num_player] := playername;
            sex[num_player] := sex_in;
            playcount[num_player] := 0;
            score[num_player] := 0;

            store_players_info;

            pw_OK := true;
            writeln;
            writeln('        User account created successfully.');
            write('        Press <Enter> to return. ');
            readln
          end
      end
  until pw_OK
end;
```

6)First page of program

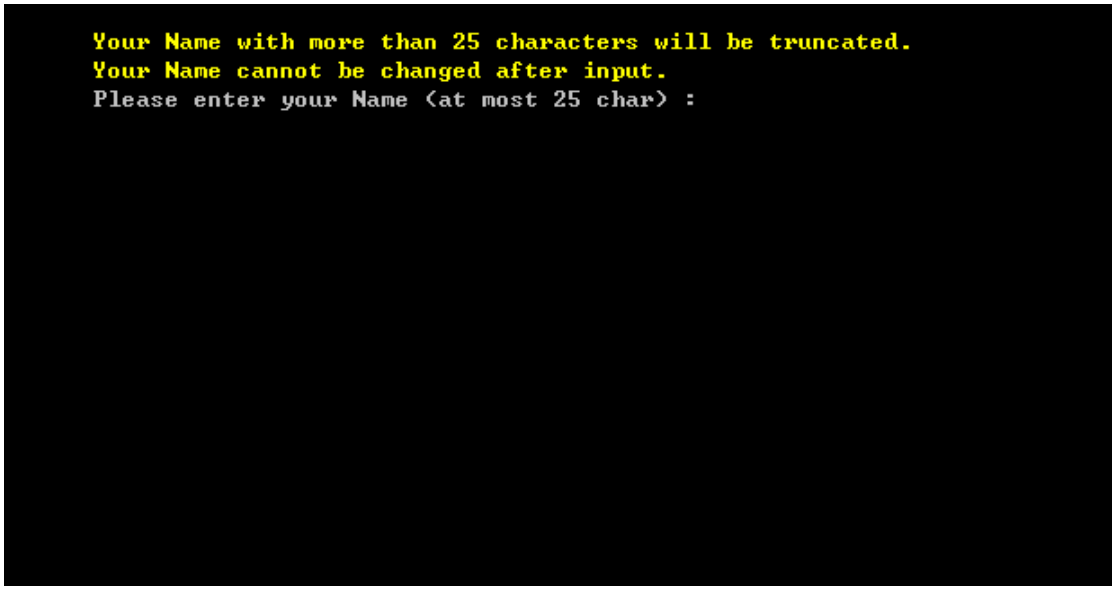
After you start the program, you will be found in the front page to choose whether you are going to register or login.

```
procedure first_page;
begin
    clrscr;
    writeln;
    writeln('                    Welcome to CSW CAL system ');
    writeln('                         Please choose:');
    writeln;
    writeln('               ------------------------------------');
    writeln;
    writeln('                    1. Login');
    writeln;
    writeln('                    2. Register');
    writeln;
    writeln('               ------------------------------------');
    writeln;
    write('                   Enter your choice: ');
    readln(Choice);
    writeln;
    case choice of
      1 : login(player_index);
      2 : create_acc;
    end;
end;
```

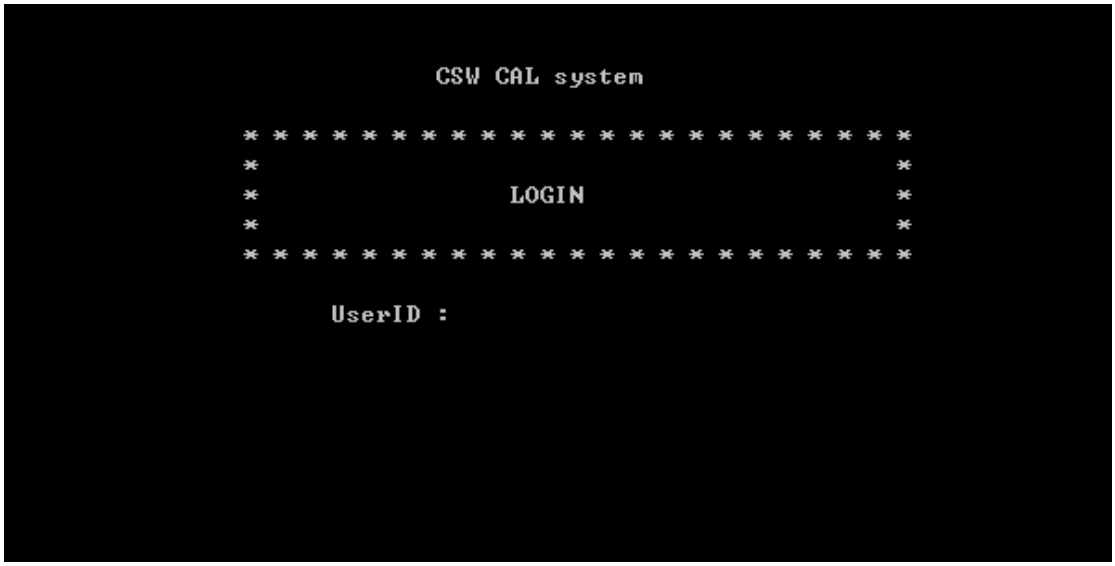
Sample:



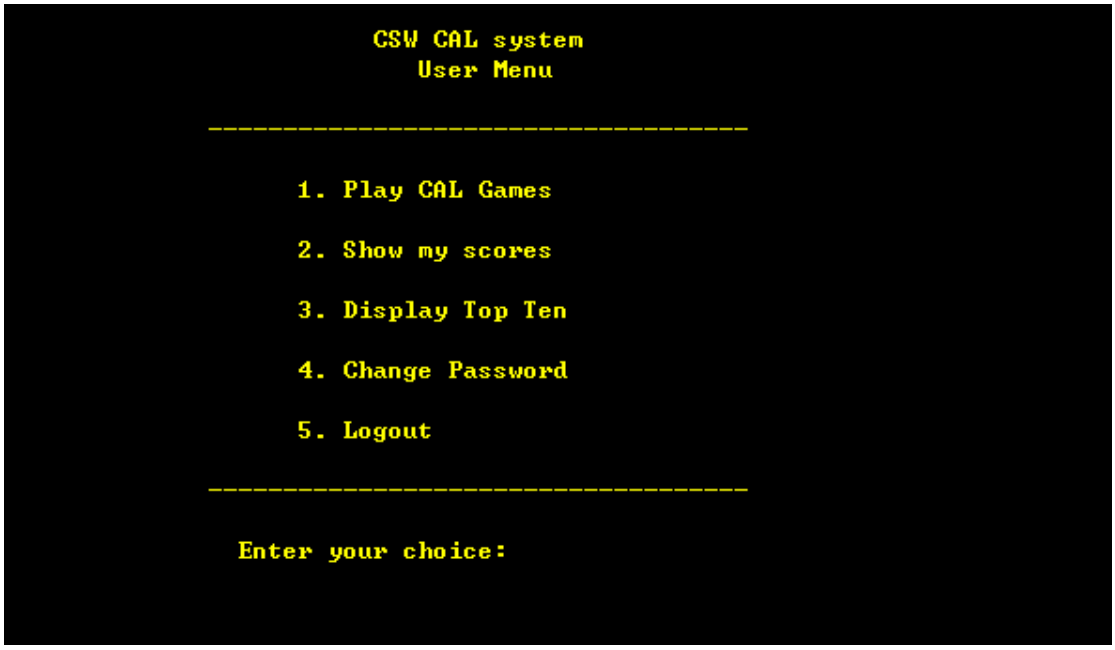
If you choose enter 2, register, you need to fill in the information and passcode to finish the register system



If you choose 1, there is a login system. Player can login through Player ID and passcode.

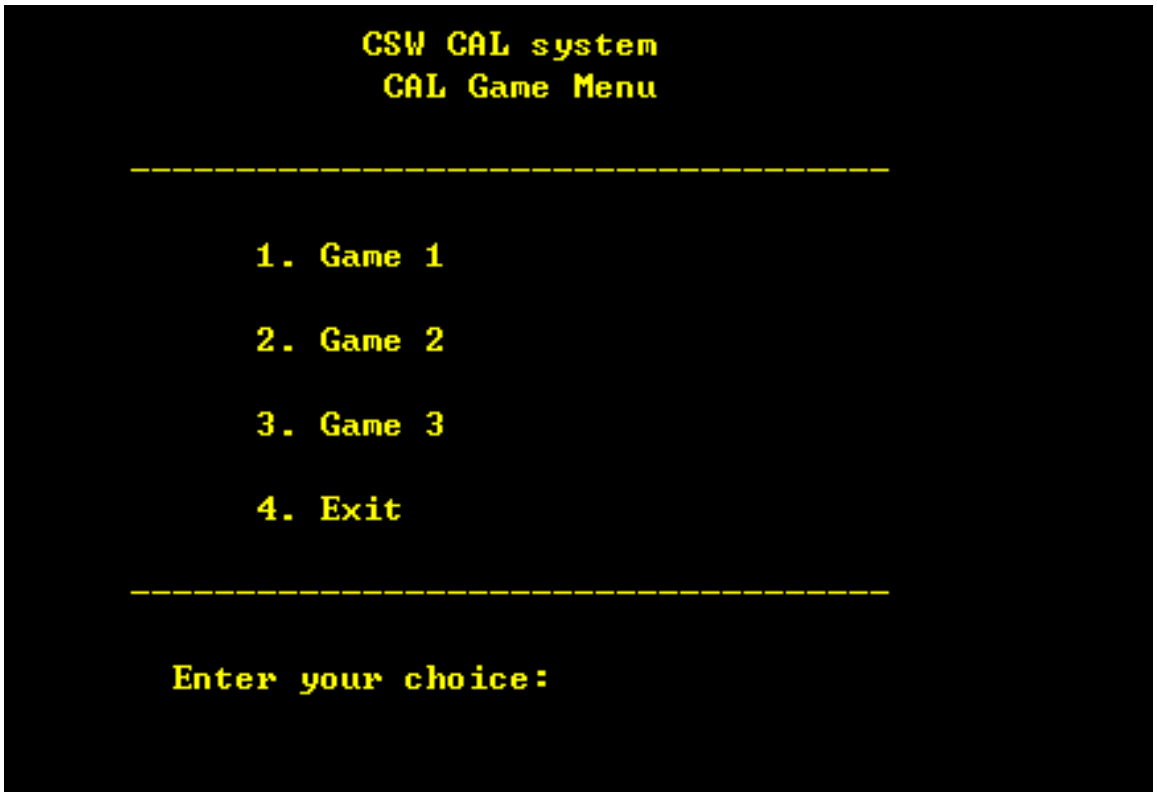


After login an register , you will enter the User Menu.

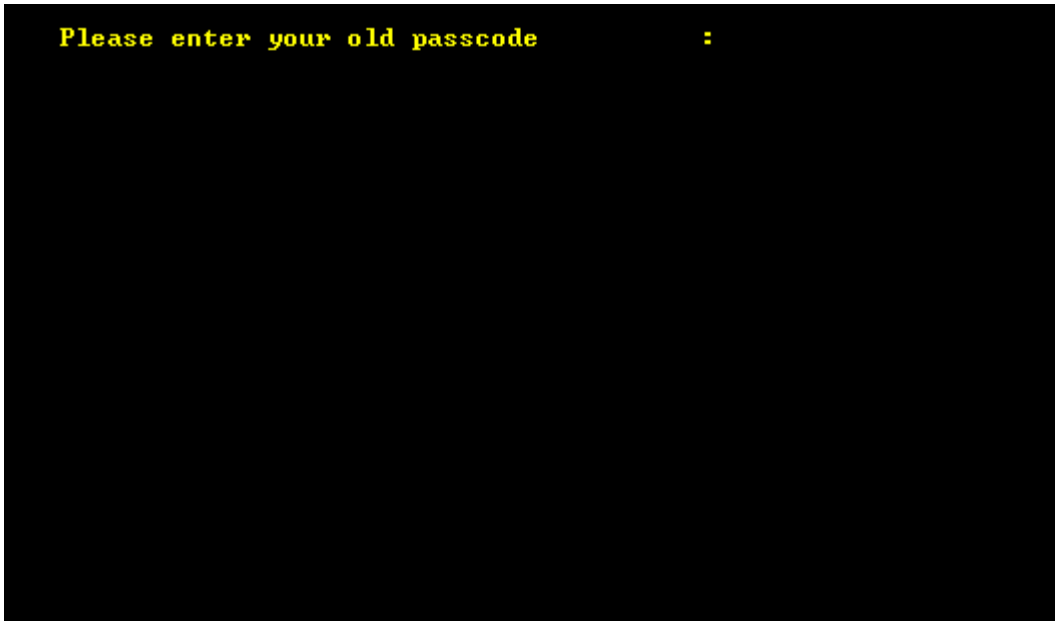


You can choose games here by enter

“1”,”2”and”3”

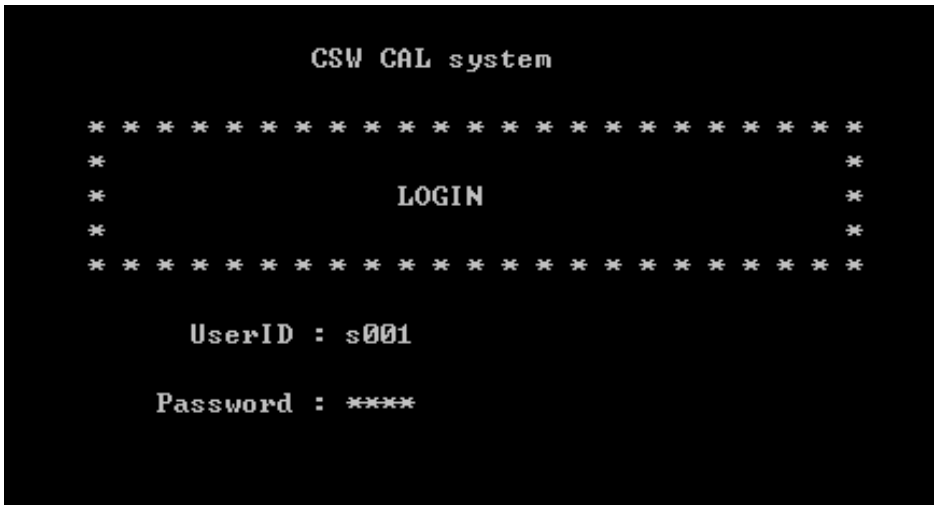


Change password in the menu by pressing “4”



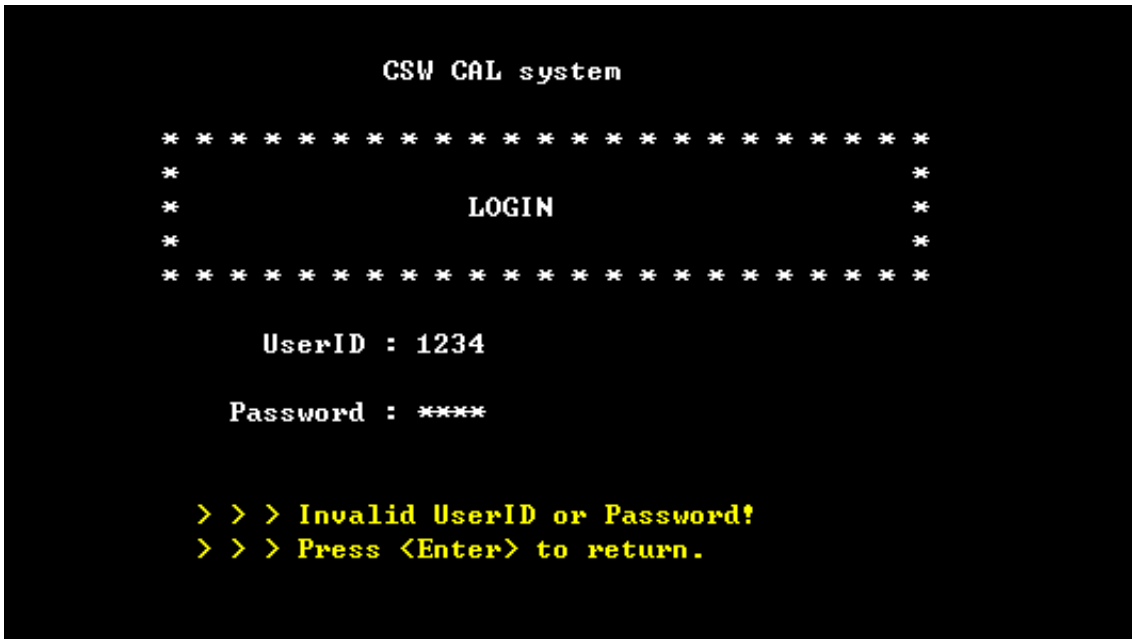
Testing

Test case1



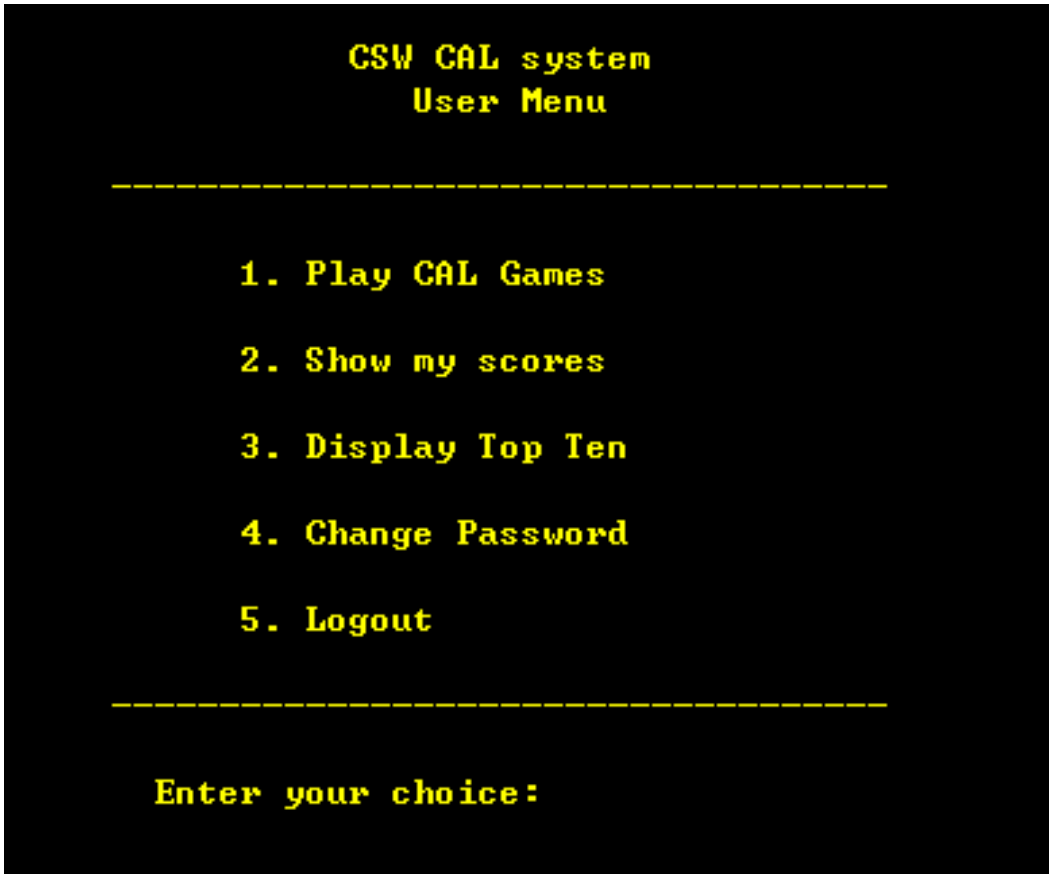
Purpose:	To check if the system can correctly identify whether the inputted data is correct or not
Input:	Existing players ID and password
Expected Output:	Go to the menu
Actual Output:	Same as the expected results.
Test Result:	Pass / No bugs found
Follow-up Action:	Nil

Test case 2



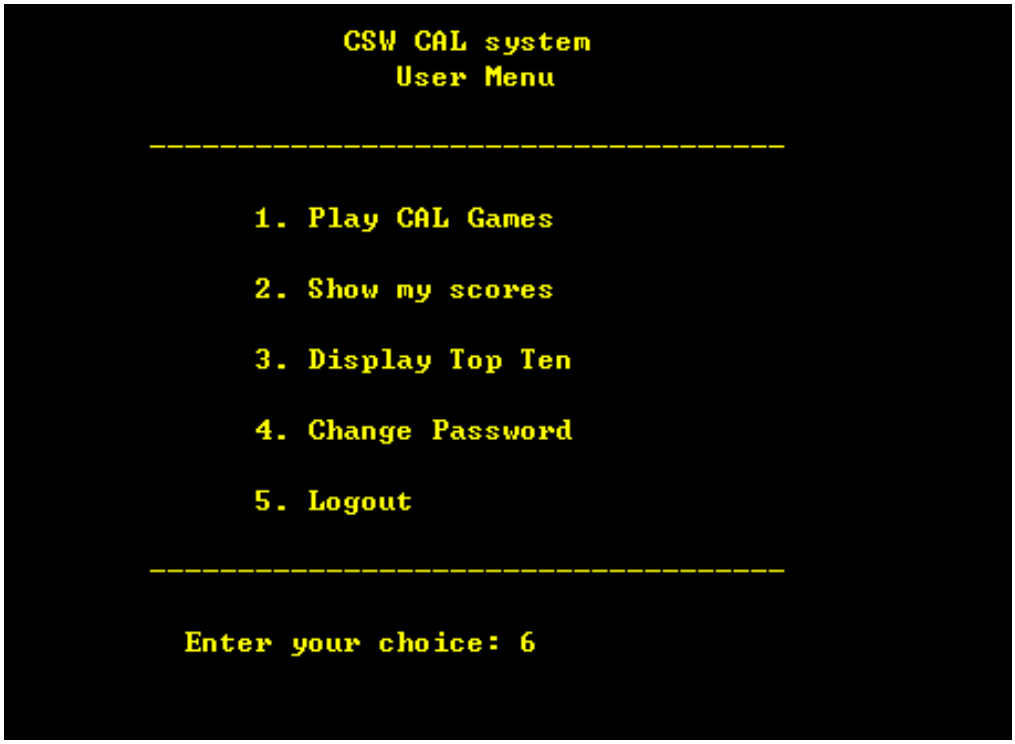
Purpose:	To check if the system can correctly identify whether the inputted data is valid or not
Input:	Fake student ID and password
Expected Output:	Invalid User ID or Password! Press<Enter> to return
Actual Output:	All actual results are the same as the expected results.
Test Result:	Pass / No bugs found
Follow-up Action:	Nil

Test case 3



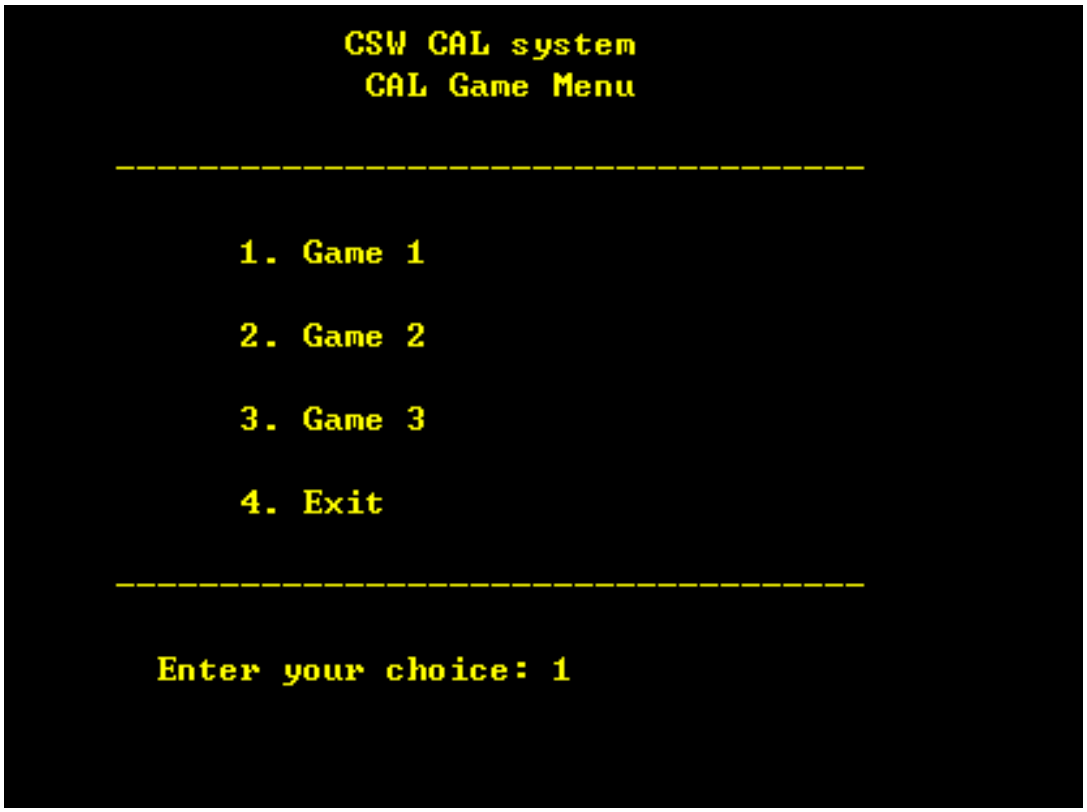
Purpose:	To check if the system can correctly identify whether the inputted data is within the range of 1-5
Input:	'1'
Expected Output:	Immediately go to the Cal games centre
Actual Output:	Allactual results are the same as the expected results.
Test Result:	Pass
Follow-up Action:	Nil

Test case 4



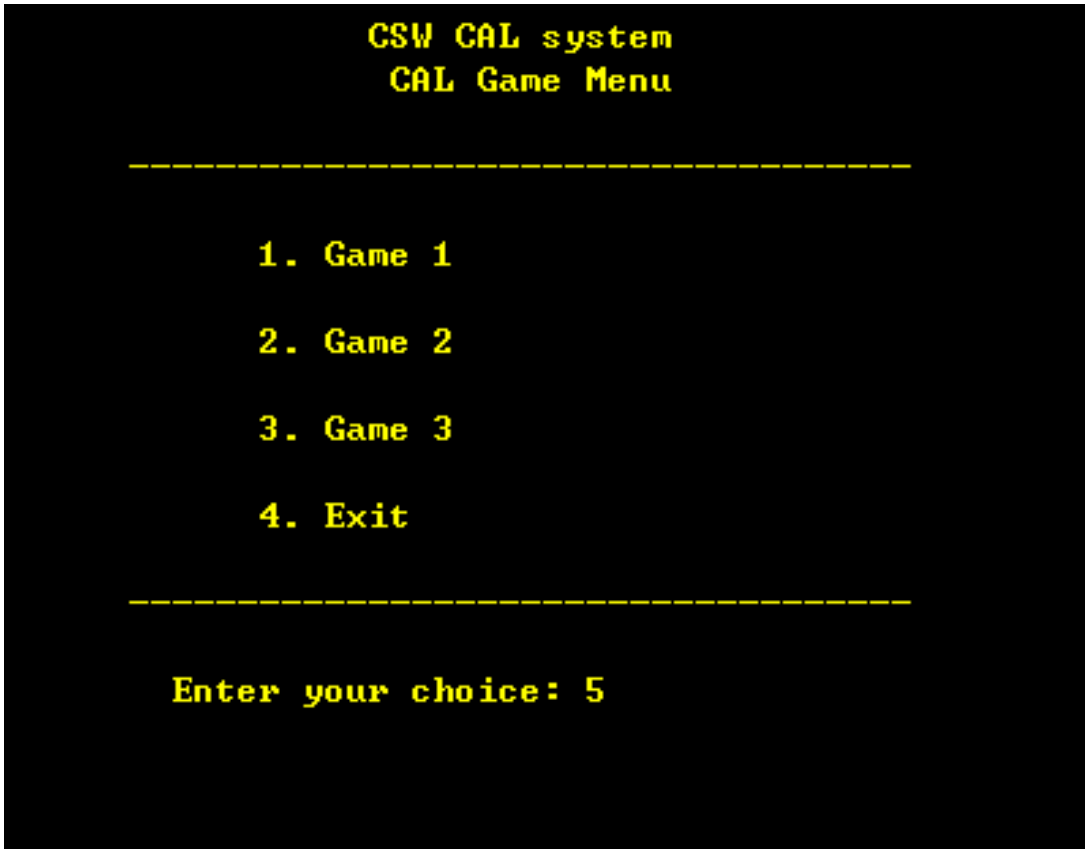
Purpose:	To check if the system can correctly identify whether the inputted data is out of range
Input:	‘6’
Expected Output:	The program should ask the user to input the data again
Actual Output:	All actual results are the same as the expected results.
Test Result:	Pass
Follow-up Action:	Nil

Test case 5



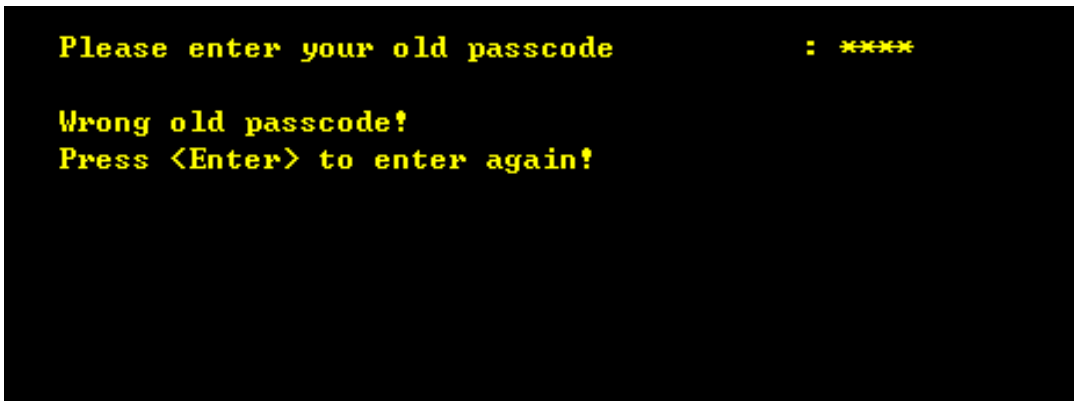
Purpose:	To check if the system can correctly identify whether the inputted data is out of range
Input:	'1'
Expected Output:	Immediately go to the Games
Actual Output:	All actual results are the same as the expected results.
Test Result:	Pass
Follow-up Action:	Nil

Test case 6



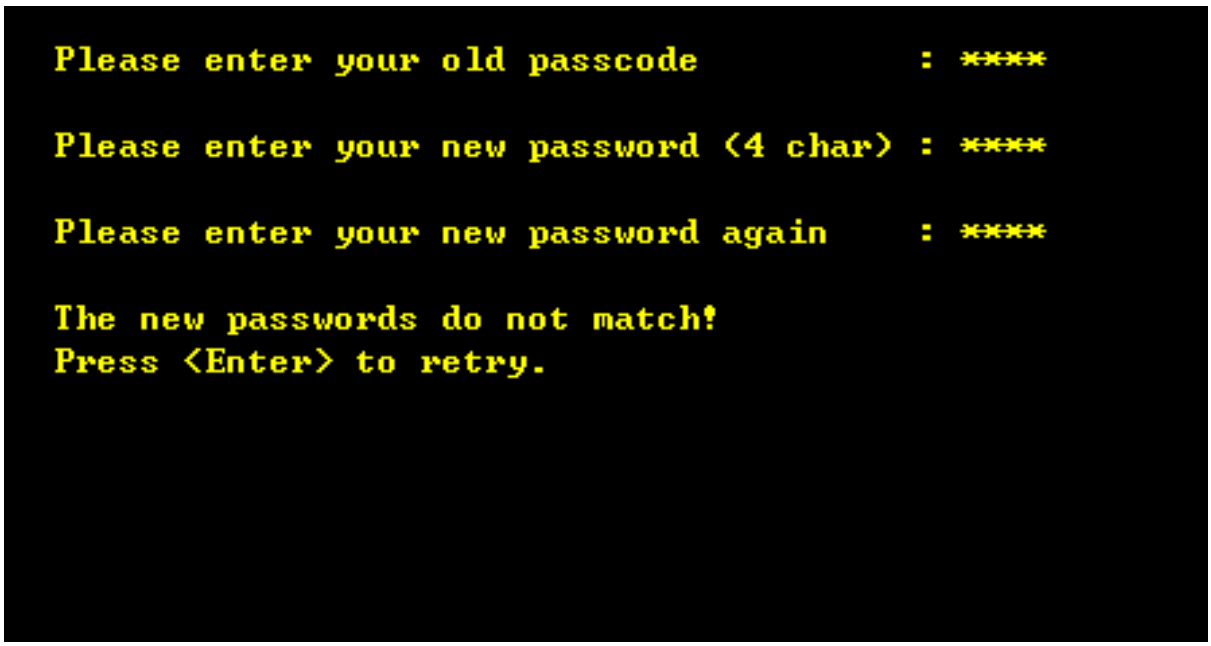
Purpose:	To check if the system can correctly identify whether the inputted data is out of range
Input:	'5'
Expected Output:	The program should ask the player to input the data again
Actual Output:	All actual results are the same as the expected results.
Test Result:	Pass
Follow-up Action:	Nil

Test case 7



Purpose:	To check if the system can correctly identify whether the inputted data is out of range
Input:	Uncorrect use passcode
Expected Output:	“Wrong old passcode!” “Press<Enter> to enter again!” The program should ask the user to input the data again
Actual Output:	All actual results are the same as the expected results.
Test Result:	Pass
Follow-up Action:	Nil

Test Case 8



Purpose:	To check if the system can correctly identify whether the inputted data is out of range
Input:	Uncorrect user new passcode
Expected Output:	“The new passwords do not match” “Press<Enter> to retry” The program should ask the user to input the data again
Actual Output:	All actual results are the same as the expected results.
Test Result:	Pass
Follow-up Action:	Nil

Evaluation

The program has a clear and clean structure and interface. It is a small game center witch provide player to enjoy the moment with friends and competitive each other. Therefore, I have face a lot of difficulties when I try to find out which game is suitable and attractive.

Besides, my program is user friendly. The player can login and register the game centre in order to play the game easily after following the clear instructions. functions of the system and it is easy to follow. The interfaces are clear and simple that users can easily find what they need.

From the perspective of my friend, who advice me to do this program. We have search a lot of resource and function to compare which is better. For example, different version of 21's. I find out that multiplayer are more entertaining

Acknowledgement

1.ICT Teacher, Mr. Chu
2. Friends and family members who have given feedbacks to me about the flaws of my program.

Appendices

Appendix1 –ProgramCode(after Testing & Evaluation)

```
program gamecentre;


uses crt;

    const

    max_player = 1000;


var

    playerid : array[1..max_player] of string[4];
```

```
    playerpw : array[1..max_player] of string[4];
    name : array[1.. max_player] of string [25];
    sex : array[1..max_player] of char;
    playcount : array[1.. max_player] of integer;
    score : array[1..max_player] of integer;
    player_index, num_player,choice : integer;


procedure read_players_info;
var
   i : integer;
   f : text;
begin
   assign(f, 'player.txt');
   reset(f);
   i := 0;
   while not eof(f) do
      begin
         i := i + 1;
```

```
            readln(f, playerid[i], playerpw[i], name[i], sex[i], playcount[i],
score[i]);
        end;
    num_player := i;
    close(f)
end;


procedure store_players_info;
var    i : integer;
         f : text;
begin
    assign(f, 'player.txt');
    rewrite(f);
    for i := 1 to num_player do
        begin
            writeln(f, playerid[i], playerpw[i], name[i], sex[i], playcount[i],' ',
score[i]);
        end;
    close(f)
```

```
end;


function GetPWord : string;
var
   S : string;
   C : Char;
begin
   S := '';
   repeat
      C := ReadKey;
      if (C <> #10) and (C <> #13) and (C <> #8) then
         begin
            S := S + C;
            write('*');
         end
      else if C = #8 then
         begin
            S[0] := Chr(Length(S) - 1);
            GotoXY(WhereX - 1, WhereY);
```

```
            write(' ');
            GotoXY(WhereX - 1, WhereY);
        end;
  until (C = #10) or (C = #13);
  GetPWord := S;
  writeLn;
end;

procedure display_player_score(player_index : integer);
 begin
  textcolor(Yellow);
  writeln(' <----------------------------------------> ');
  writeln(' <----------------------------------------> ');
  writeln;
  writeln(' *UserID: ', playerid[player_index]:5);
  writeln;
  writeln(' *Name : ', name[player_index]:5);
  writeln;
  writeln(' *Total Playcount : ', playcount[player_index]);
```

```
    writeln;
    writeln(' *Your total score : ', score[player_index]);
    writeln;
    writeln(' <---------------------------------------------> ');

writeln('wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwww');
    writeln;
    textcolor(Yellow);
    write(' Press <Enter> to return. ');
    readln
end;

procedure change_password(player_index : integer);
var
    oldpass, newpass1, newpass2 : string;
    pwchanged : boolean;
begin
    pwchanged := false;
```

```
repeat
   clrscr;
   writeln;
   write('              Please enter your old passcode                  : ');
   oldpass := GetPword;
   if oldpass <> playerpw[player_index] then
      begin
         writeln;
         writeln('              Wrong old passcode!');
         write('              Press <Enter> to enter again! ');
         readln
      end
   else
      begin
         writeln;
         write('              Please enter your new password (4 char) : ');
         newpass1 := GetPword;
         if length(newpass1) <> 4 then
            begin
```

```
                writeln;
                writeln('              The length of password must be 4!');
                write('              Press <Enter> to retry. ');
                readln
              end
            else
              begin
                writeln;
                write('              Please enter your new password
again       : ');
                newpass2 := GetPword;
                if newpass1 <> newpass2 then
                  begin
                    writeln;
                    writeln('              The new passwords do not
match!');
                    write('              Press <Enter> to retry. ');
                    readln
                  end
```

```
                else
                    begin
                        playerpw[player_index] := newpass1;
                        store_players_info;
                        pwchanged := true;
                        writeln;
                        writeln('              Password changed.');
                        write('              Press <Enter> to return. ');
                        readln
                    end
                end
            end
    until pwchanged
end;



procedure game1(player_index : integer);
var
    numCor : integer;
```

```
    usercard1, usercard2, dealercard1, dealercard2, usertotal, dealertotal,
userwins, dealerwins, deal : integer;


  begin
    textcolor(15);
    clrscr;
    randomize;
    dealerwins := 0;
    userwins:= 0;
    deal := 1;
    writeln( '>----------------------<');
    writeln( '    You Guess What? ' );
    writeln( '>----------------------<');
    usercard1 := trunc(random * 13)+1;
    usercard2 := trunc(random * 13)+1;
    dealercard1 := trunc(random *13)+1;
    dealercard2 := trunc(random *13)+1;
      usertotal := (usercard1 + usercard2);
    dealertotal := (dealercard1 + dealercard2);
```

```
    writeln( '-------------');
    writeln( ' Deal Number ' , deal );
    writeln( '-------------');
    writeln;
    writeln( ' Your Cards: ' , usercard1:2 , ' and ' , usercard2:2, ' Total ' ,
usertotal );
    writeln;
    writeln( ' Dealers Cards: ' , dealercard1:2 , ' and ' , dealercard2:2, '
Total ' , dealertotal );
    writeln;
    if (usertotal > 21) then
      begin
        writeln( ' You Bust The Dealer wins ');
        numCor := numCor -1


      end;
      if (dealertotal > 21)then
      begin
        writeln( ' The Dealer Bust You win ');
```

```
      numCor := numCor + 1
  end;
if (usertotal < dealertotal)then
    begin
    writeln( ' The Dealer wins ' );
    numCor := numCor -1
  end;
if (usertotal < dealertotal)then
  inc(dealerwins);
  if (usertotal > dealertotal)then
  begin
  writeln(' You win ');
  numCor := numCor + 1
  end;
if (usertotal > dealertotal)then
  inc(userwins);

if (usertotal = dealertotal)then
writeln( ' A Draw ' );
```

```
    writeln;
    writeln(' Dealer wins ' , dealerwins );
    writeln;
    writeln(' Your Wins ' , userwins );
    writeln;
    score[player_index] := score[player_index] + numCor;
    store_players_info;
    writeln;
    textcolor(15);
    writeln('Would you like to play MASTERMIND again? (y,n): ');
    readln;
end;


procedure play_gamecentre(player_index : integer);
var
    choice : integer;
begin
    repeat
        clrscr;
```

```
writeln;
writeln('                                CSW CAL system ');
writeln('                                 CAL Game Menu');
writeln;
writeln('                    ----------------------------------');
writeln;
writeln('                              1. Game 1');
writeln;
writeln('                              2. Game 2 ');
writeln;
writeln('                              3. Game 3');
writeln;
writeln('                              4. Exit');
writeln;
writeln('                    ----------------------------------');
writeln;
write('                     Enter your choice: ');
readln(Choice);
writeln;
```

```
        case choice of
            1 : game1(player_index);
        end;
    until choice = 4;
end;


procedure main_menu(player_index : integer);
var
    choice : integer;
begin
    repeat
        clrscr;
        writeln;
        textcolor(Yellow);
        writeln('                                    CSW CAL system ');
        writeln('                                        User Menu');
        writeln;
        writeln('                          ----------------------------------');
        writeln;
```

```
writeln('                    1. Play CAL Games');
writeln;
writeln('                    2. Show my scores ');
writeln;
writeln('                    3. Display Top Ten');
writeln;
writeln('                    4. Change Password');
writeln;
writeln('                    5. Logout');
writeln;
writeln('                  ------------------------------------');
writeln;
write('                     Enter your choice: ');
readln(Choice);
writeln;
case choice of
   1 : play_gamecentre(player_index);
   2 : display_player_score(player_index);
```

```
            4 : change_password(player_index);
        end;
    until choice = 5;
end;



procedure login(var player_index : integer);
var
    loginid, password : string;
    found : boolean;
    i : integer;
begin
    clrscr;
    writeln;
    writeln;
```

```
    writeln('                                    CSW CAL system
');
    writeln;
    writeln('                    * * * * * * * * * * * * * * * * * * * * * * *
*           ');
    writeln('                    *
*           ');
    writeln('                    *                    LOGIN
*           ');
    writeln('                    *
*           ');
    writeln('                    * * * * * * * * * * * * * * * * * * * * * * *
*           ');
    writeln;
    write('                              UserID : ');
    readln(loginid);
    writeln;
    write('                            Password : ');
    password := GetPword;
```

```
writeln;
writeln;
found := false;
i := 0;
while (i < num_player) and (not found) do
   begin
      i := i + 1;
      if (loginid = playerid[i]) and (password = playerpw[i]) then
         begin
            found := true;
            player_index := i
         end
   end;
if not found then
   begin
      player_index := 0;
      textcolor(yellow);
      writeln(':20,'> > > Invalid UserID or Password!');
      write(':20,'> > > Press <Enter> to return.');
```

```
            textcolor(white);
            readln
        end
    else
        main_menu(player_index)
end;

procedure create_acc;
var
    loginid, playername, password1, password2 : string;
    sex_in : char;
    id_OK, pw_OK : boolean;
    i : integer;
begin
    clrscr;
    writeln;
    textcolor(Yellow);
    writeln('              Your Name with more than 25 characters will be
truncated.');
```

```
writeln('              Your Name cannot be changed after input.');
textcolor(LightGray);
write('              Please enter your Name (at most 25 char) : ');
readln(playername);
repeat
   writeln;
   write('              Please enter your Gender (M or F) : ');
   readln(sex_in)
until sex_in in ['M', 'F'];

id_OK := false;
repeat
   clrscr;
   writeln;
   writeln('              Welcome ', playername, '!');
   writeln;
   write('              Please enter your PlayerID (4 char) : ');
   readln(loginid);
   if length(loginid) <> 4 then
```

```
    begin
      writeln;
      writeln('              The length of PlayerID must be 4!');
      write('              Press <Enter> to retry. ');
      readln
    end
  else
    begin
      id_OK := true;
      for i := 1 to num_player do
        if loginid = playerid[i] then
          begin
            writeln;
            writeln('              The PlayerID has been used!');
            write('              Press <Enter> to retry. ');
            readln;
            id_OK := false
          end
    end
```

```
until id_OK;


pw_OK := false;
repeat
   clrscr;
   writeln;
   writeln('              Welcome ', playername, '!');
   writeln;
   writeln('              Your UserID is ', loginid);
   writeln;
   write('              Please enter your password (4 char) : ');
   password1 := GetPword;
   if length(password1) <> 4 then
      begin
         writeln;
         writeln('              The length of password must be 4!');
         write('              Press <Enter> to retry. ');
         readln
      end
```

```
            else
              begin
                writeln;
                write('              Please enter your password again       : ');
                password2 := GetPword;
                if password1 <> password2 then
                  begin
                    writeln;
                    writeln('              The passwords do not match!');
                    write('              Press <Enter> to retry. ');
                    readln
                  end
                else
                  begin
                    num_player := num_player + 1;
                    playername := playername + '
';
                    playername := copy(playername, 1, 25);
```

```
                    playerid[num_player] := loginid;
                    playerpw[num_player] := password1;
                    name[num_player] := playername;
                    sex[num_player] := sex_in;
                    playcount[num_player] := 0;
                    score[num_player] := 0;

                    store_players_info;

                    pw_OK := true;
                    writeln;
                    writeln('              User account created
successfully.');
                    write('              Press <Enter> to return. ');
                    readln
                  end
              end
   until pw_OK
end;
```

```
procedure first_page;
begin
    clrscr;
    writeln;
    writeln('                    Welcome to CSW CAL system ');
    writeln('                        Please choose:');
    writeln;
    writeln('                ----------------------------------');
    writeln;
    writeln('                    1. Login');
    writeln;
    writeln('                    2. Register');
    writeln;
    writeln('                ----------------------------------');
    writeln;
    write('                    Enter your choice: ');
    readln(Choice);
    writeln;
```

```
        case choice of
            1 : login(player_index);
            2 : create_acc;
        end;
end;


begin
    read_players_info;
    repeat
        first_page;
    until false;
    readln
end.
```