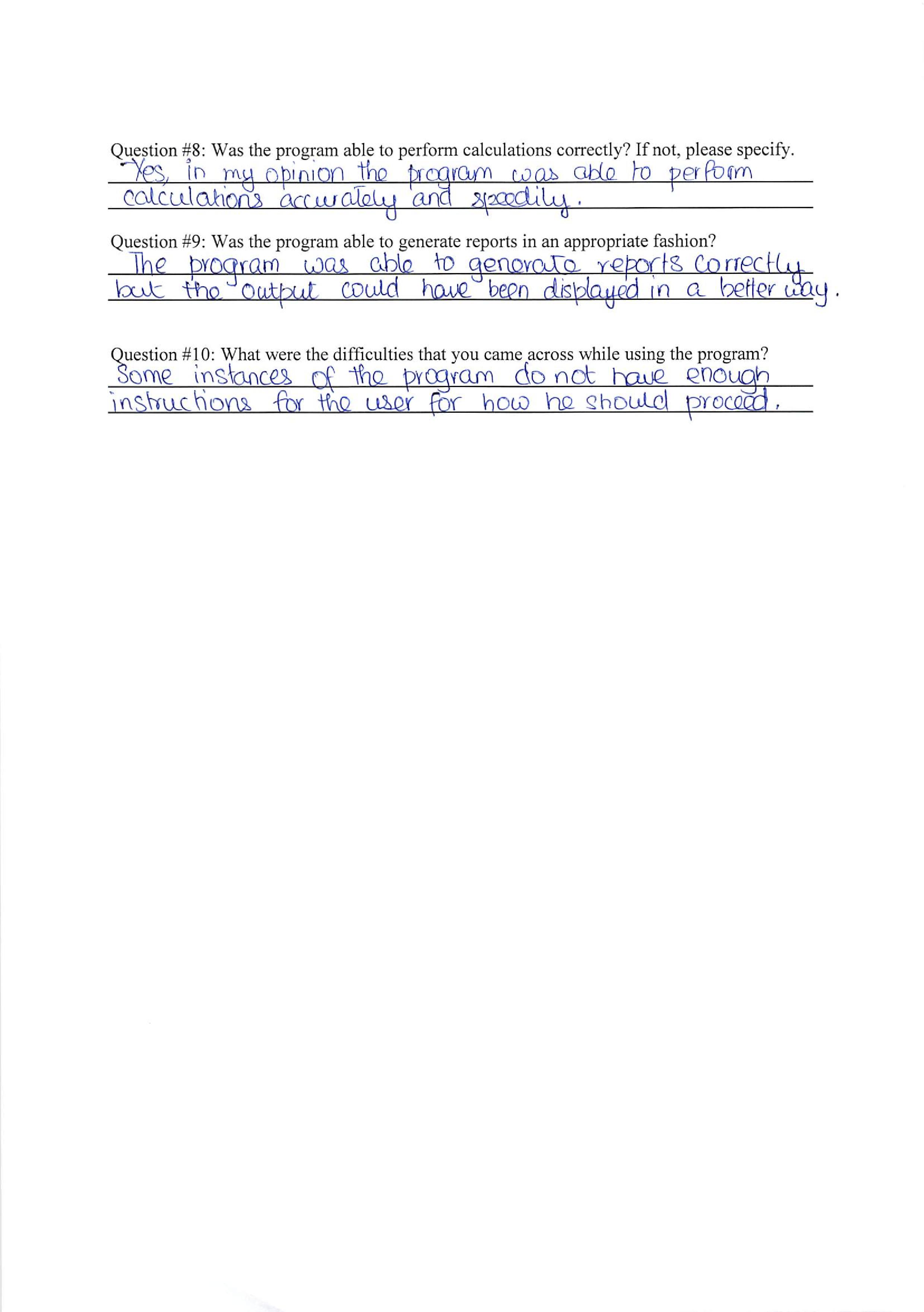
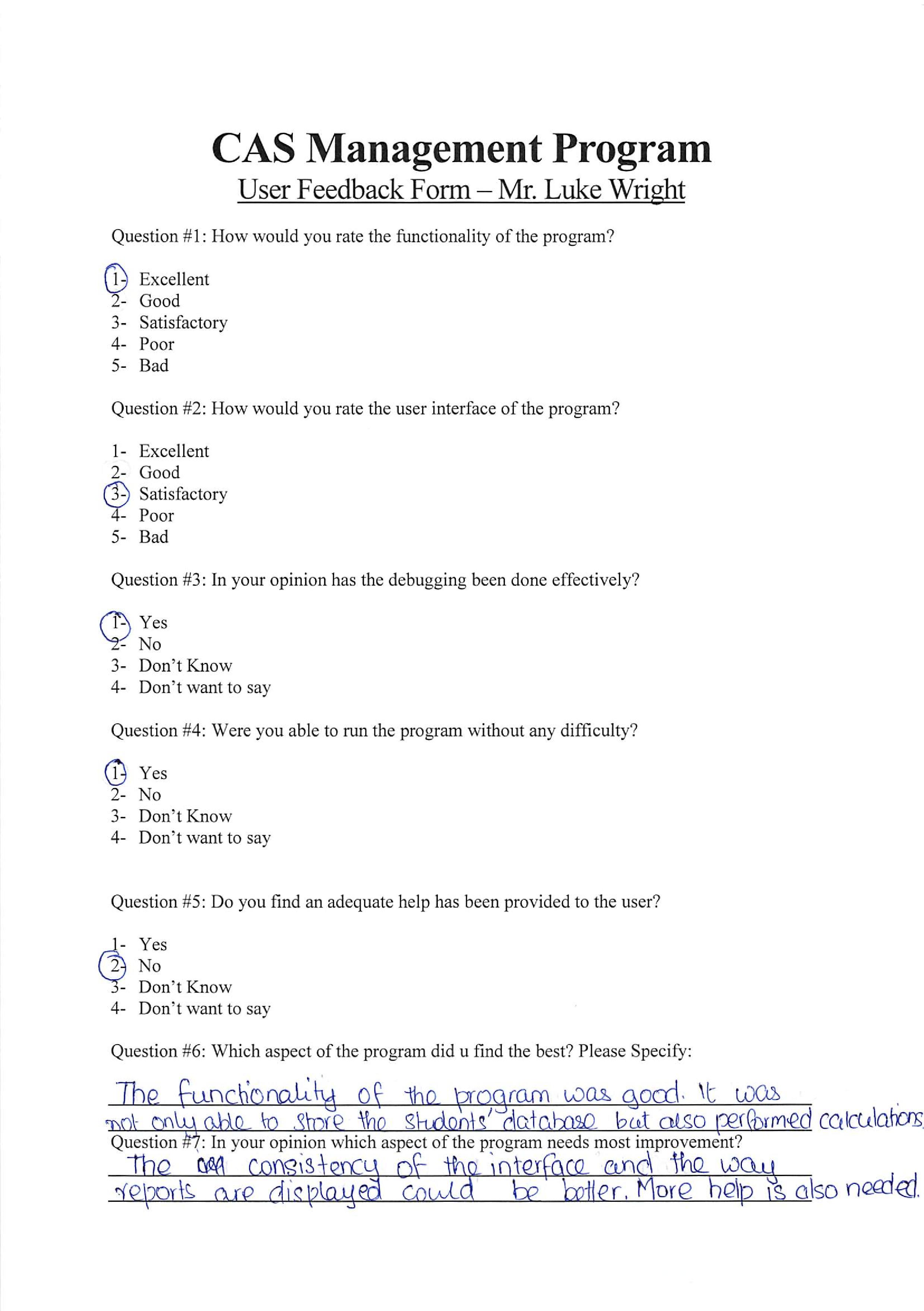
# Appendix: User Feedback & Source Code

## User Feedback



## Source Code

import java.io.\*;

import java.util.Scanner;

class program {

//MAIN METHOD STARTS HERE

public static void main(String args[]) {

System.out.println(

"\tWelcome to CAS Manager Lite\nFor more information please contact 'Mateen Ahmed'.\nFor 'Press any key to continue' please press enter!"

);

init();

int i = 0;

boolean loginSuccessful = false;

while (loginSuccessful == false) {

if (i != 0)

System.out.println(

"Username or Password has been entered incorrectly! Please try again..."

);

loginSuccessful = login();

i = i + 1;

}

makeLine();

System.out.println(

"Login successful!");

control();

}

//=========================================================================

//DIRECTORY & CREDENTIALS INITIALIZING METHOD STARTS HERE

public static void init() {

Scanner Input = new Scanner(

System. in );

File mainDirectory = new File(

"C:\\AppData");

if (!mainDirectory.exists()) {

mainDirectory.mkdir();

File loginDirectory = new File(

"C:\\AppData\\Login");

loginDirectory.mkdir();

try {

RandomAccessFile file =

new RandomAccessFile(

"C:\\AppData\\Login\\credentials.dat",

"rw");

file.seek(0);

makeLine();

System.out.print(

"The program is executed for the first time!\n"

);

makeLine();

System.out.print(

"Please specifiy a username:\t"

);

String userName = Input

.nextLine();

System.out.print(

"Please specify a password:\t"

);

String passWord = Input

.nextLine();

userName = userName +

"\n";

passWord = passWord +

"\n";

file.writeBytes(

userName);

file.writeBytes(

passWord);

} catch (Exception e) {

System.out.println(

"Some IO error occured."

);

init();

}

}

}

//=========================================================================

//LOGIN METHOD STARTS HERE - CHECKS CREDENTIALS AND ALLOWS/DENIES ACCESS ACCORDINGLY

public static boolean login() {

Scanner Input = new Scanner(

System. in );

boolean match = false;

System.out.println(

"\n\t\tLOGIN MENU!");

makeLine();

System.out.print(

"Enter your username:\t");

String userName = Input.nextLine();

userName = userName.toLowerCase();

userName = userName.trim();

System.out.print(

"Enter your password:\t");

String passWord = Input.nextLine();

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Login\\credentials.dat",

"rw");

file.seek(0);

String currentUsername =

file.readLine();

currentUsername =

currentUsername.trim();

currentUsername =

currentUsername.toLowerCase();

String currentPassword =

file.readLine();

if (userName.equals(

currentUsername) &&

passWord.equals(

currentPassword))

match = true;

} catch (Exception e) {

System.out.println(

"Some I/O Error Occured!"

);

}

return match;

}

//==========================================================================

//MAIN MENU CONTROLLING METHOD STARTS HERE

public static void control() {

directoryChecker();

Scanner Input = new Scanner(

System. in );

int myMainChoice = 0;

while (myMainChoice != 6) {

System.out.println(

"\n\t\tMAIN MENU!");

makeLine();

System.out.println("");

System.out.println(

"Please select from the following menu:\n\n1)Add new student\n2)Add student hours\n3)View details about existing students\n4)Perform Calculations/Generate Reports\n5)Exit"

);

System.out.println("");

makeLine();

System.out.print(

"Enter choice:\t");

myMainChoice = Input.nextInt();

switch (myMainChoice) {

case 1:

addStudent();

break;

case 2:

addHours();

break;

case 3:

viewStudents();

break;

case 4:

calculate();

break;

case 5:

System.out.println(

"Exiting program now.."

);

makeLine();

System.exit(1);

break;

default:

System.out.println(

"Not a valid entry! "

);

makeLine();

control();

break;

}

}

} //==========================================================================

//DIRECTORY CHECIKING METHOD STARTS HERE - CHECKS THE DIRECTORY FOR THE STUDENTS FILES AND CREATES IT IF NOT FOUND

public static void directoryChecker() {

File studentDirectory = new File(

"C:\\AppData\\Students");

if (!studentDirectory.exists()) {

try {

studentDirectory.mkdir();

RandomAccessFile file =

new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

long FileSize = file.length();

file.seek(FileSize);

file.writeInt(1);

file.writeBytes(

"----- \n");

file.writeBytes(

"----- \n");

file.close();

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

long File2Size = file2.length();

file2.seek(File2Size);

file2.writeInt(00);

file2.writeInt(00);

file2.writeInt(00);

file2.writeInt(00);

file2.close();

} catch (Exception e) {

System.out.println(

"Some I/O Error Occured!"

);

}

}

}

//=========================================================================

//ADD STUDENTS METHOD STARTS HERE

public static void addStudent() {

int condition = 1;

Scanner Input = new Scanner(

System. in );

makeLine();

System.out.println(

"\n\t\tMAKE NEW STUDENT");

while (condition == 1) {

makeLine();

String firstName = "";

String lastName = "";

try {

RandomAccessFile file =

new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

file.seek(file.length() -

26);

int studentID = file.readInt() +

1;

System.out.println(

"Student ID:\t" +

studentID);

while (firstName.equals(

"") || lastName

.equals("")) {

makeLine();

System.out.print(

"Enter first name:\t"

);

firstName = Input.nextLine();

System.out.print(

"Enter first name:\t"

);

firstName = Input.nextLine();

System.out.print(

"Enter last name:\t"

);

lastName = Input.nextLine();

}

int firstLength = 10 -

firstName.length();

for (int i = 0; i <

firstLength; i++)

firstName =

firstName + " ";

int lastLength = 10 -

lastName.length();

for (int i = 0; i <

lastLength; i++)

lastName = lastName +

" ";

int creativity = 0;

int action = 0;

int service = 0;

System.out.println("");

makeLine();

System.out.print(

"Do you want to add more? Press 1 for more or other number to stop:\t"

);

condition = Input.nextInt();

int student =

studentCount();

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

file.seek((studentCount() \*

26) + 0);

file.writeInt(studentID);

file.writeBytes(

firstName + "\n");

file.writeBytes(

lastName + "\n");

file.close();

file2.seek((

studentCount() \*

16) + 0);

file2.writeInt(

studentID);

file2.writeInt(

creativity);

file2.writeInt(action);

file2.writeInt(service);

file2.close();

} catch (Exception e) {

System.out.println(

"Some I/O error occured!"

);

}

}

}

//=========================================================================

//ADD HOURS METHOD STARTS HERE

public static void addHours() {

int condition = 1;

Scanner Input = new Scanner(

System. in );

makeLine();

System.out.println(

"\n\t\tADD HOURS");

while (condition == 1) {

makeLine();

System.out.print(

"Enter student ID to add hours to:\t"

);

int studentID = Input.nextInt();

studentID = studentID - 1;

int found = 0;

while (found != 1) {

try {

RandomAccessFile file =

new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

int id1[] = new int[

studentCount()];

for (int z = 0; z <

studentCount() -

1; z++) {

file.seek(26 \*

z);

id1[z] = file.readInt();

if (studentID ==

id1[z])

found = 1;

}

if (found == 0) {

System.out.print(

"Sorry the user does not exist!\nPlease reenter the ID:\t"

);

studentID =

Input.nextInt();

studentID =

studentID -

1;

}

} catch (Exception e) {

System.out.println(

"Some I/O Error Occured!"

);

}

}

makeLine();

System.out.print(

"Enter new creativity hours:\t"

);

int creativity = Input.nextInt();

System.out.print(

"Enter new action hours:\t\t"

);

int action = Input.nextInt();

System.out.print(

"Enter new service hours:\t"

);

int service = Input.nextInt();

try {

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

file2.seek(file2.length());

file2.writeInt(

studentID);

file2.writeInt(

creativity);

file2.writeInt(action);

file2.writeInt(service);

makeLine();

} catch (Exception e) {

makeLine();

System.out.println(

"Some I/O Error Occured!"

);

}

System.out.print(

"Do you want to add more? Press 1 to edit more or other number to exit:\t"

);

condition = Input.nextInt();

}

}

//=========================================================================

//VIEW STUDENT METHOD STARTS HERE

public static void viewStudents() {

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tVIEW STUDENTS");

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

System.out.println("");

System.out.print(

"Student ID\tFirst Name\tLast Name\tCreativity\tAction\t\tService\n"

);

System.out.println( "========================================================================================="

);

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 1; g <

studentCount(); g++) {

System.out.print(" " +

(id1[g]));

System.out.print(

"\t\t " + fName[g]

);

System.out.print("\t " +

lName[g]);

System.out.print(

"\t " + c[g]);

System.out.print(

"\t\t " + a[g]);

if (s[g] <= 9) {

System.out.print(

"\t " +

s[g]);

} else System.out.print(

"\t " + s[

g]);

System.out.println("");

}

makeLongLine();

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

}

//=========================================================================

//CALCULATING METHOD STARTS HERE - PERFORMS ALL THE CALCULATIONS

public static void calculate() {

Scanner Input = new Scanner(

System. in );

System.out.println(

"\n\t\tPERFORM CALCULATIONS"

);

makeLine();

System.out.println(

"Please select from the following calculations:\nCalculation #1: Total CAS Hours per student\n" +

"Calculation #2: Students who have failed to complete CAS\nCalculation #3: Students who completed CAS successfully\n" +

"Calculation #4: Students with under certain criteria \nCalculation #5: Highest CAS - top 3\n" +

"Calculation #6: All statistics for one student\n"

);

makeLine();

System.out.print(

"Please enter your choice (1-5):\t"

);

int myChoice = Input.nextInt();

switch (myChoice) {

case 1:

totalCAS();

control();

break;

case 2:

failingCAS();

control();

break;

case 3:

completedCAS();

control();

break;

case 4:

CAScriteria();

control();

break;

case 5:

topCAS();

control();

break;

case 6:

oneStudent();

control();

break;

}

}

//=========================================================================

//Total CAS hours counting method starts here

public static void totalCAS() {

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tREPORT");

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

System.out.println("");

System.out.print(

"Student ID\tFirst Name\tLast Name\tTotal CAS\n"

);

System.out.println(

"========================================================================================="

);

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 1; g <

studentCount(); g++) {

int total = 0;

System.out.print(" " +

(id1[g]));

System.out.print(

"\t\t " + fName[g]

);

System.out.print("\t " +

lName[g]);

total = c[g] + a[g] + s[

g];

System.out.println(

"\t " + total);

}

makeLongLine();

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

}

//=========================================================================

//Students failed CAS method begins here

public static void failingCAS() {

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tREPORT");

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

System.out.println("");

System.out.print(

"Student ID\tFirst Name\tLast Name\tCAS Status\n"

);

System.out.println(

"========================================================================================="

);

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 1; g <

studentCount(); g++) {

if (c[g] < 50 || a[g] <

50 || s[g] < 50) {

System.out.print(

" " + (id1[g])

);

System.out.print(

"\t\t " +

fName[g]);

System.out.print(

"\t " + lName[g]

);

System.out.println(

" Failing CAS!"

);

}

}

makeLongLine();

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

}

//=========================================================================

//Students completed CAS method begins here

public static void completedCAS() {

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tREPORT");

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

System.out.println("");

System.out.print(

"Student ID\tFirst Name\tLast Name\tCAS Status\n"

);

System.out.println(

"========================================================================================="

);

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 1; g <

studentCount(); g++) {

if (c[g] >= 50 && a[g] >=

50 && s[g] >= 50) {

System.out.print(

" " + (id1[g])

);

System.out.print(

"\t\t " + fName[

g]);

System.out.print(

"\t " + lName[g]

);

System.out.println(

" Completed CAS!"

);

}

}

makeLongLine();

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

}

//=========================================================================

//Students under certain CAS hours method starts here

public static void CAScriteria() {

Scanner Input = new Scanner(

System. in );

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tREPORT");

makeLine();

System.out.print(

"Please enter the upper CAS hours limit:\t"

);

int criteria = Input.nextInt();

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

System.out.println("");

System.out.print(

"Student ID\tFirst Name\tLast Name\tTotal CAS\n"

);

System.out.println(

"========================================================================================="

);

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 1; g <

studentCount(); g++) {

int total = 0;

total = c[g] + a[g] + s[

g];

if (total < criteria) {

System.out.print(

" " + (id1[g])

);

System.out.print(

"\t\t " + fName[

g]);

System.out.print(

"\t " + lName[g]

);

System.out.println(

" Hours < " +

criteria);

}

if (total == criteria) {

System.out.print(

"\t" + (id1[g])

);

System.out.print(

"\t\t\t\t " +

fName[g]);

System.out.print(

"\t " + lName[

g]);

System.out.println(

"Hours = " +

criteria);

}

}

makeLongLine();

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

}

//=========================================================================

//Students with 3 highest CAS hours method begins here

public static void topCAS() {

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tREPORT");

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

System.out.println("");

System.out.print(

"Student ID\tFirst Name\tLast Name\tRanking\n"

);

System.out.println(

"========================================================================================="

);

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int totals[] = new int[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 0; g <

studentCount(); g++) {

int total = 0;

total = c[g] + a[g] + s[

g];

totals[g] = total;

}

for (int i = studentCount() -

1; i > 0; i--) {

for (int j = 0; j < i; j++) {

if (totals[j] >

totals[j + 1]) {

int tempID =

id1[j];

id1[j] = id1[j +

1];

id1[j + 1] =

tempID;

String tempFname =

fName[j];

fName[j] =

fName[j + 1];

fName[j + 1] =

tempFname;

String tempLname =

lName[j];

lName[j] =

lName[j + 1];

lName[j + 1] =

tempLname;

int tempTotal =

totals[j];

totals[j] =

totals[j +

1];

totals[j + 1] =

tempTotal;

}

}

}

for (int i = studentCount() -

1; i >= 0; i--) {

System.out.print(" " +

(id1[i]));

System.out.print(

"\t\t " + fName[i]);

System.out.print("\t " +

lName[i]);

if (i == studentCount() -

1)

System.out.println(

" Highest");

if (i == studentCount() -

2)

System.out.println(

" 2nd Highest"

);

if (i == studentCount() -

3) {

System.out.println(

" 3rd Highest"

);

makeLongLine();

break;

}

}

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

}

//=========================================================================

//One student statistics calculating method starts here

public static void oneStudent() {

Scanner Input = new Scanner(

System. in );

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"rw");

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"rw");

System.out.println(

"\n\t\tREPORT");

makeLine();

System.out.print(

"Please enter the ID of the student:\t"

);

int studentID = Input.nextInt();

makeLine();

System.out.println(

"NOTE: Student I D #1 is reserved! Records start from #2!"

);

makeLine();

file2.seek(0);

file.seek(0);

int id1[] = new int[

studentCount()];

String fName[] = new String[

studentCount()];

String lName[] = new String[

studentCount()];

int totals[] = new int[

studentCount()];

int ID1 = 0;

String FName = "";

String LName = "";

for (int h = 0; h <

studentCount(); h++) {

ID1 = file.readInt();

FName = file.readLine();

LName = file.readLine();

id1[h] = ID1;

fName[h] = FName;

lName[h] = LName;

}

int id2[] = new int[

hoursCount()];

int c[] = new int[

hoursCount()];

int a[] = new int[

hoursCount()];

int s[] = new int[

hoursCount()];

int ID2 = 0;

int cr = 0;

int ac = 0;

int se = 0;

for (int j = 1; j <

hoursCount() + 1; j++) {

ID2 = file2.readInt();

cr = file2.readInt();

ac = file2.readInt();

se = file2.readInt();

id2[ID2] = ID2;

c[ID2] = c[ID2] + cr;

a[ID2] = a[ID2] + ac;

s[ID2] = s[ID2] + se;

}

for (int g = 1; g <

studentCount(); g++) {

totals[g] = c[g] + a[g] +

s[g];

}

int found = binarySearch(

id1, studentID,

studentCount());

if (found == -1)

System.out.print(

"Sorry! Student with ID: " +

studentID +

" not found");

else {

System.out.println(

"Student ID:\t" +

id1[found]);

System.out.println(

"First Name:\t" +

fName[found]);

System.out.println(

"Last Name:\t" +

lName[found]);

System.out.println(

"Creativity hours:\t" +

c[found]);

System.out.println(

"Action hours:\t" +

a[found]);

System.out.println(

"Service hours:\t" +

s[found]);

System.out.println(

"Total hours:\t" +

totals[found]);

if (c[found] >= 50 && a[

found] >= 50 &&

s[found] >= 50)

System.out.println(

"CAS status:\tCAS Complete!"

);

else

System.out.println(

"CAS status:\tCAS Inomplete!"

);

}

makeLine();

} catch (Exception e) {

System.out.println(

"Some I/O error occured in printing!"

);

}

} //=========================================================================

//Binary searching method starts here

public static int binarySearch(int id1[],

int targetID, int IDsize) {

int middle, low, high;

boolean found = false;

low = 0;

high = IDsize - 1;

middle = -1;

while (high >= low && !found) {

middle = (low + high) / 2;

if (targetID < id1[middle])

high = middle - 1;

else if (targetID > id1[

middle])

low = middle + 1;

else

found = true;

}

if (found)

return middle;

else

return -1;

}

//=========================================================================

//Student counting method - Counts number of students in the students file

public static int studentCount() {

try {

RandomAccessFile file = new RandomAccessFile(

"C:\\AppData\\Students\\students.dat",

"r");

int records = (int)(file.length() /

26);

file.close();

return records;

} catch (Exception e) {

System.out.println(

"Some I/O error occured!"

);

return -1;

}

} //=========================================================================

//Hours counting method - Counts number of hours in the hours file

public static int hoursCount() {

try {

RandomAccessFile file2 =

new RandomAccessFile(

"C:\\AppData\\Students\\hours.dat",

"r");

int records = (int)(file2.length() /

16);

file2.close();

return records;

} catch (Exception e) {

System.out.println(

"Some I/O error occured!"

);

return -1;

}

}

//=========================================================================

//SIMPLE LINE DRAWING METHOD STARTS HERE

public static void makeLine() {

System.out.println(

"==================================================="

);

}

//SIMPLE LONG LINE DRAWING METHOD STARTS HERE

public static void makeLongLine() {

System.out.println(

"========================================================================================="

);

}

}