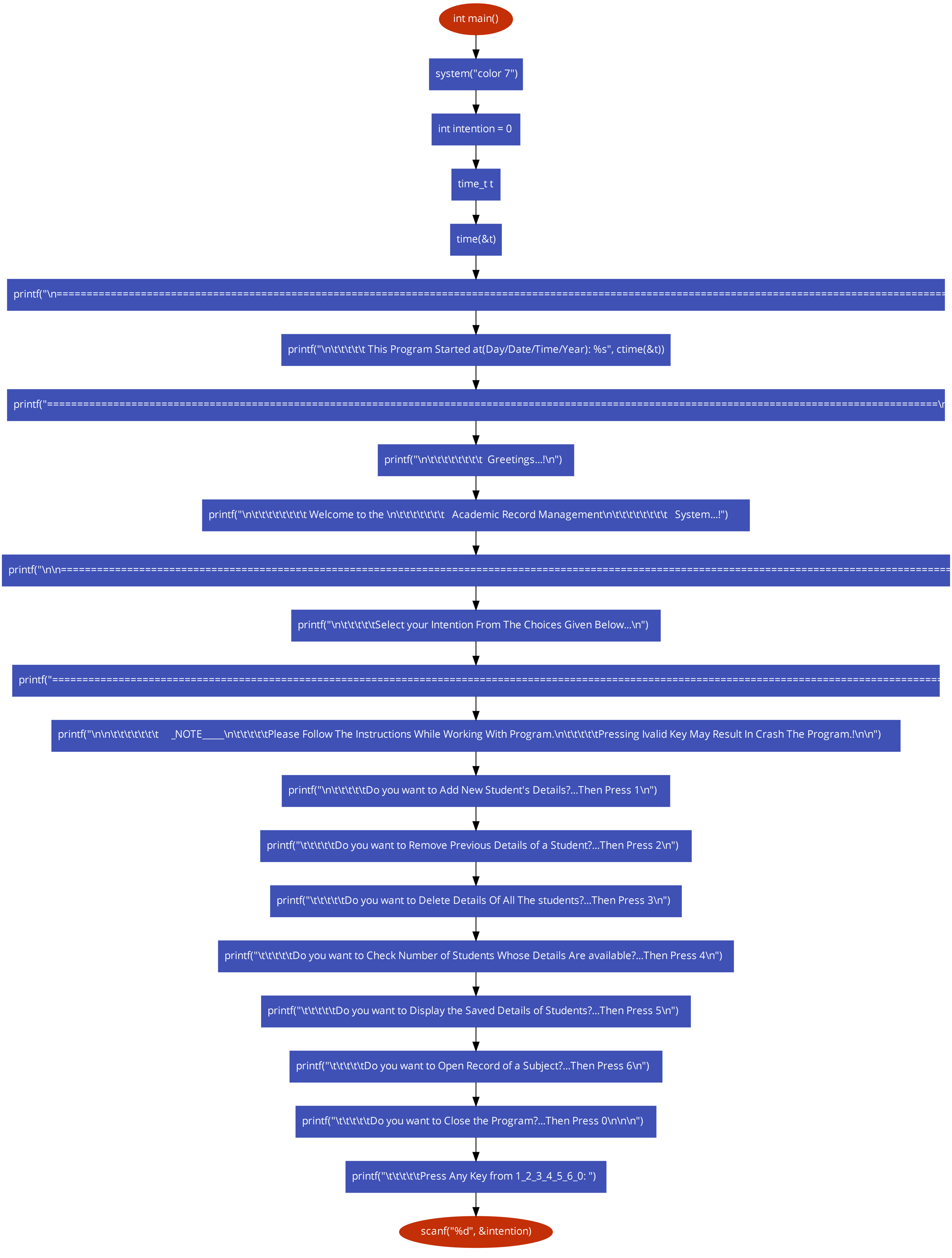
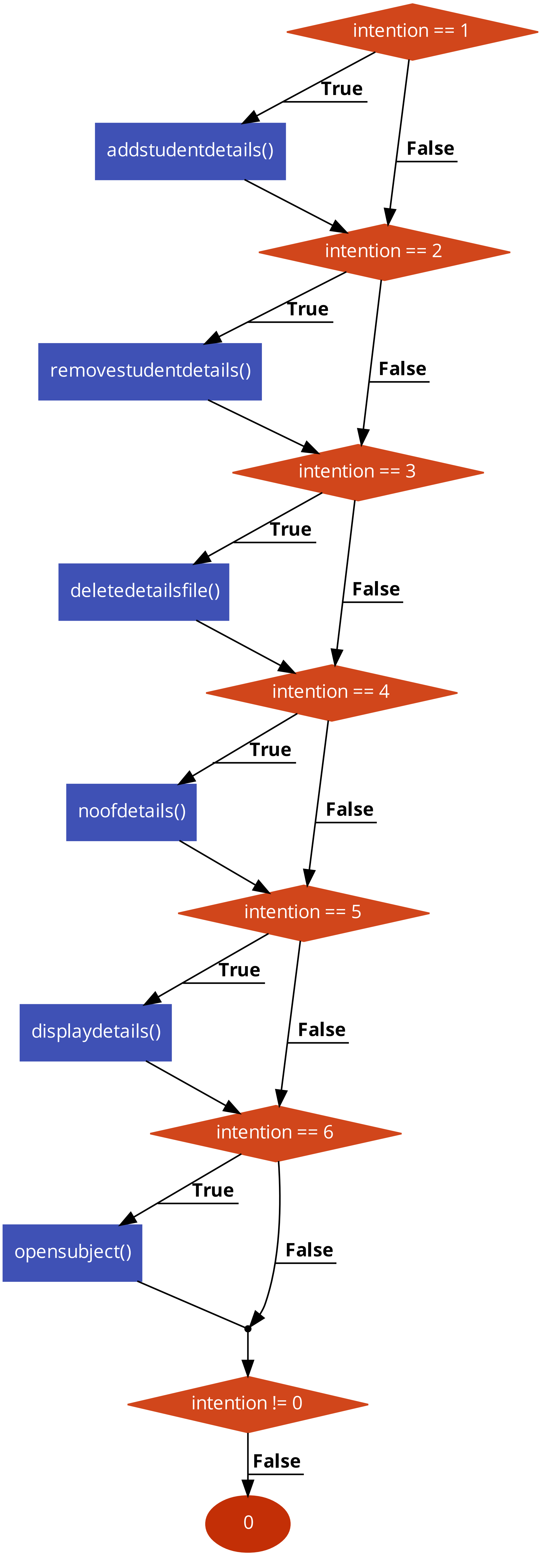
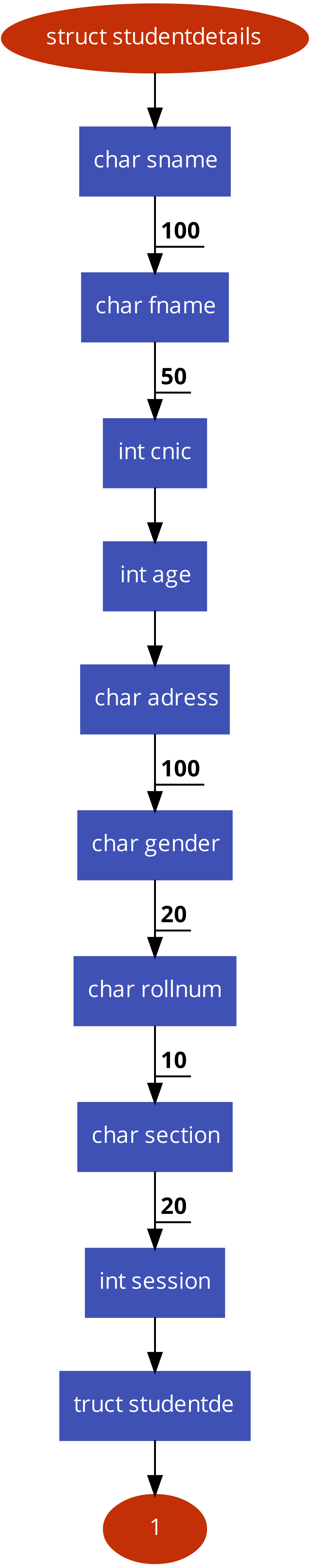
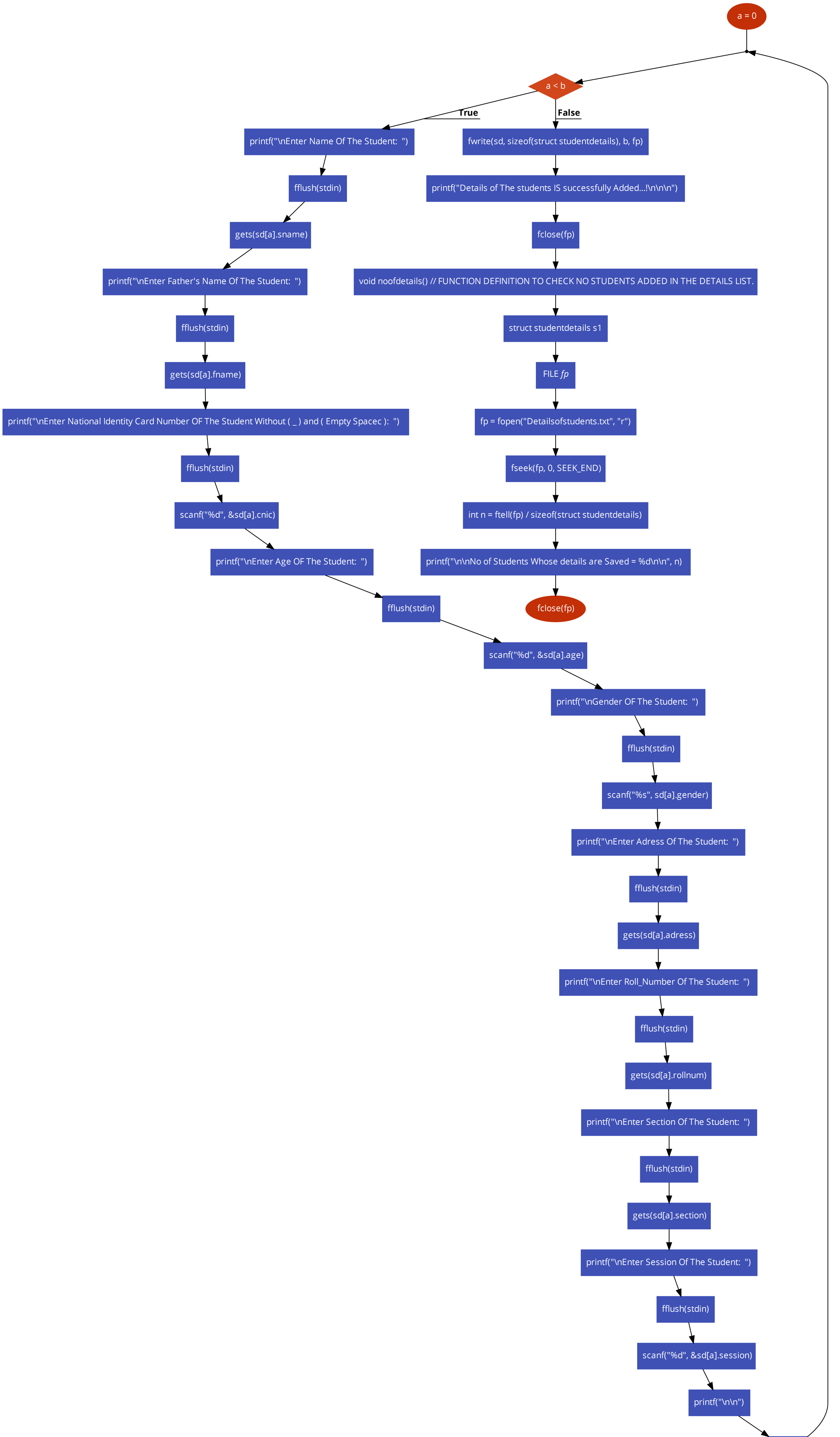
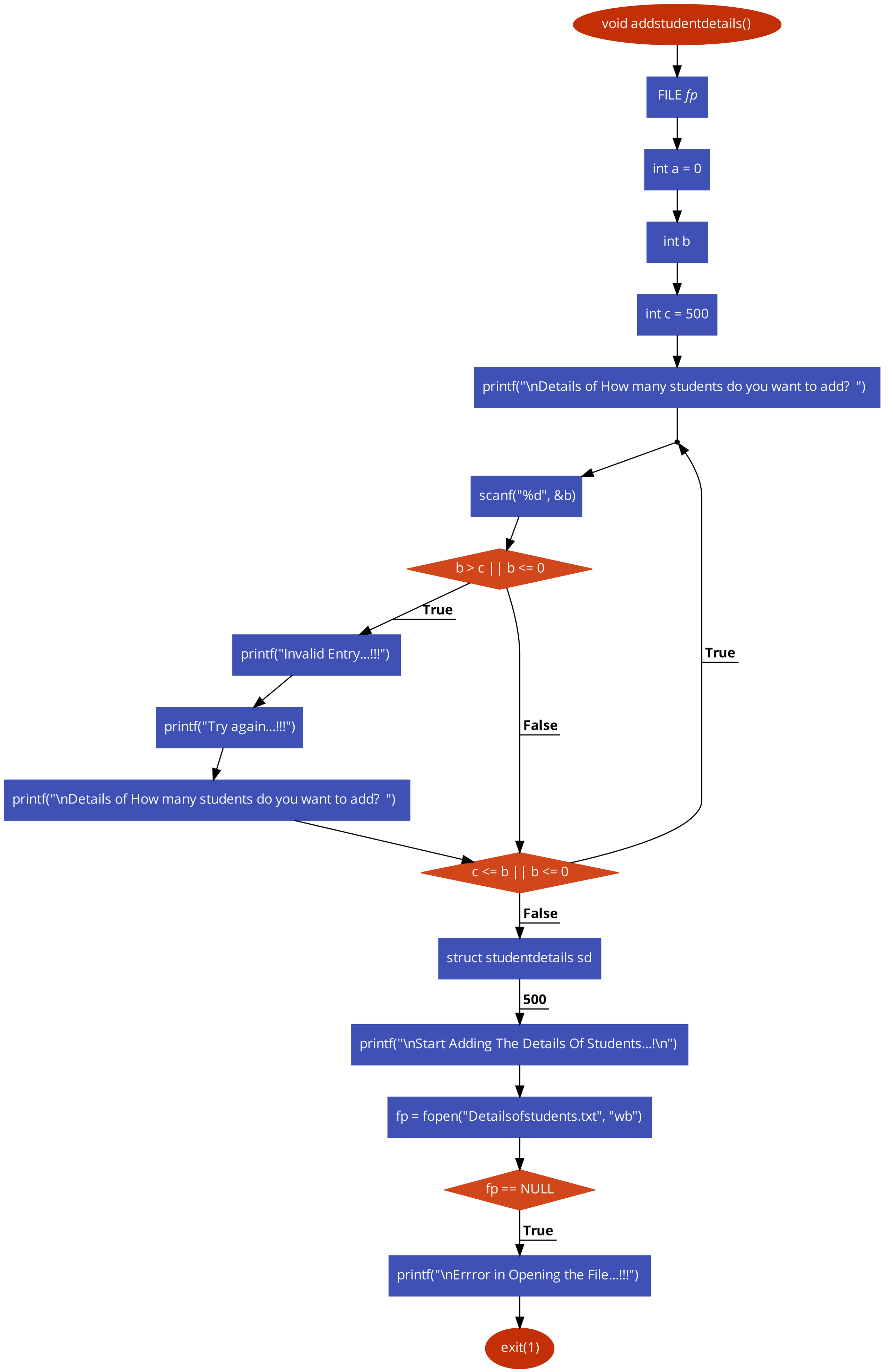
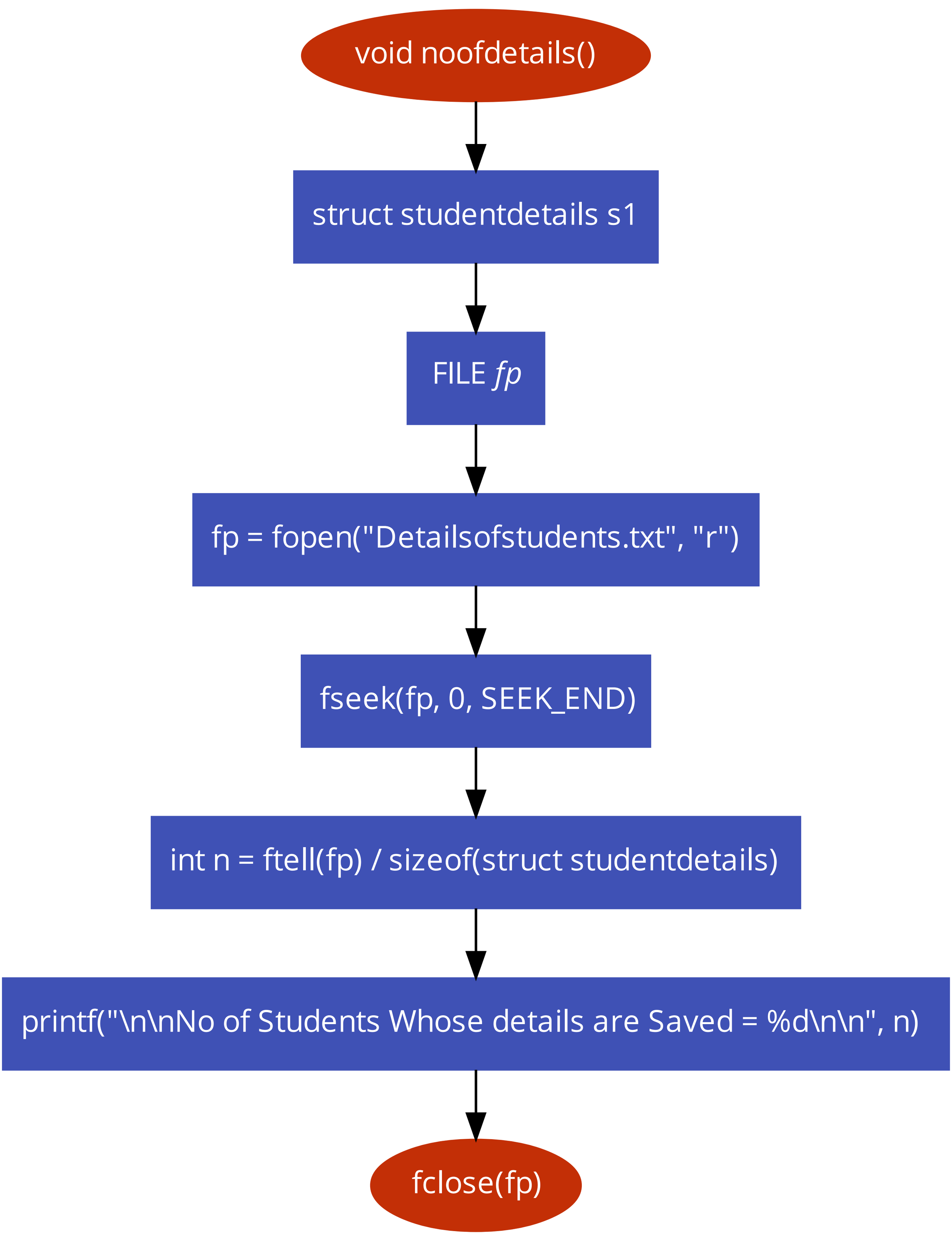

int main()
system("color 7")
int intention = 0
time_t t
time(&t)
printf("\n==========
printf("\n\t\t\t\t\t\t This Program Started at(Day/Date/Time/Year): %s", ctime(&t))
printf("==========\n
printf("\n\t\t\t\t\t\t\t\t\t Greetings...!\n")
printf("\n\t\t\t\t\t\t\t\t\t Welcome to the \n\t\t\t\t\t\t\t\t   Academic Record Management\n\t\t\t\t\t\t\t\t   System...!")
printf("\n\n==========
printf("\n\t\t\t\t\tSelect your Intention From The Choices Given Below...\n")
printf("==========
printf("\n\n\t\t\t\t\t\t   _NOTE____\n\t\t\t\t\tPlease Follow The Instructions While Working With Program.\n\t\t\t\t\tPressing Ivalid Key May Result In Crash The Program.!\n\n")
printf("\n\t\t\t\tDo you want to Add New Student's Details?...Then Press 1\n")
printf("\t\t\t\tDo you want to Remove Previous Details of a Student?...Then Press 2\n")
printf("\t\t\t\tDo you want to Delete Details Of All The students?...Then Press 3\n")
printf("\t\t\t\tDo you want to Check Number of Students Whose Details Are available?...Then Press 4\n")
printf("\t\t\t\tDo you want to Display the Saved Details of Students?...Then Press 5\n")
printf("\t\t\t\tDo you want to Open Record of a Subject?...Then Press 6\n")
printf("\t\t\t\tDo you want to Close the Program?...Then Press 0\n\n\n")
printf("\t\t\t\tPress Any Key from 1_2_3_4_5_6_0: ")
scanf("%d", &intention)

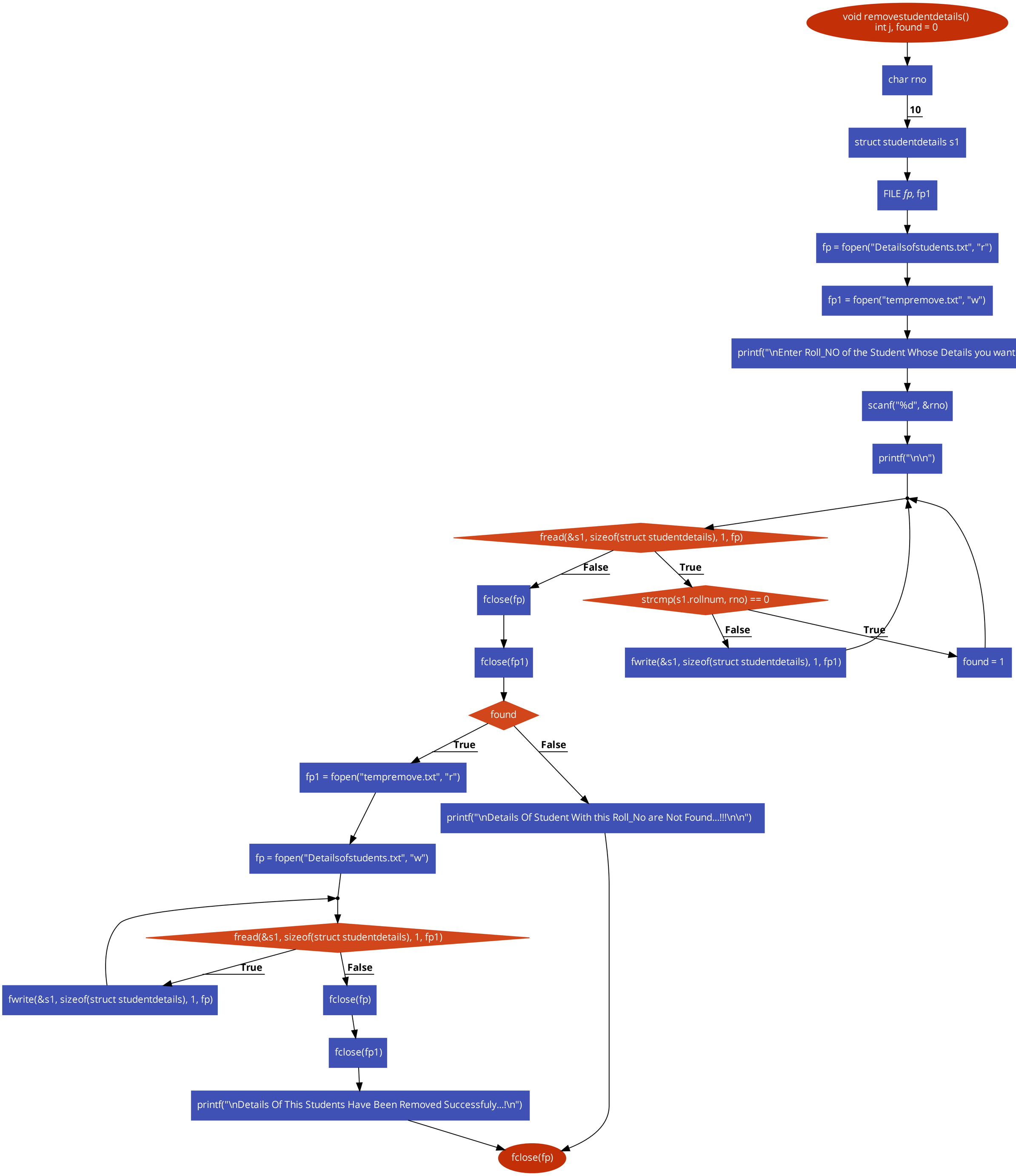

intention == 1
True
addstudentdetails()
False
intention == 2
True
removestudentdetails()
False
intention == 3
True
deletedetailsfile()
False
intention == 4
True
noofdetails()
False
intention == 5
True
displaydetails()
False
intention == 6
True
opensubject()
False
intention != 0
False
0

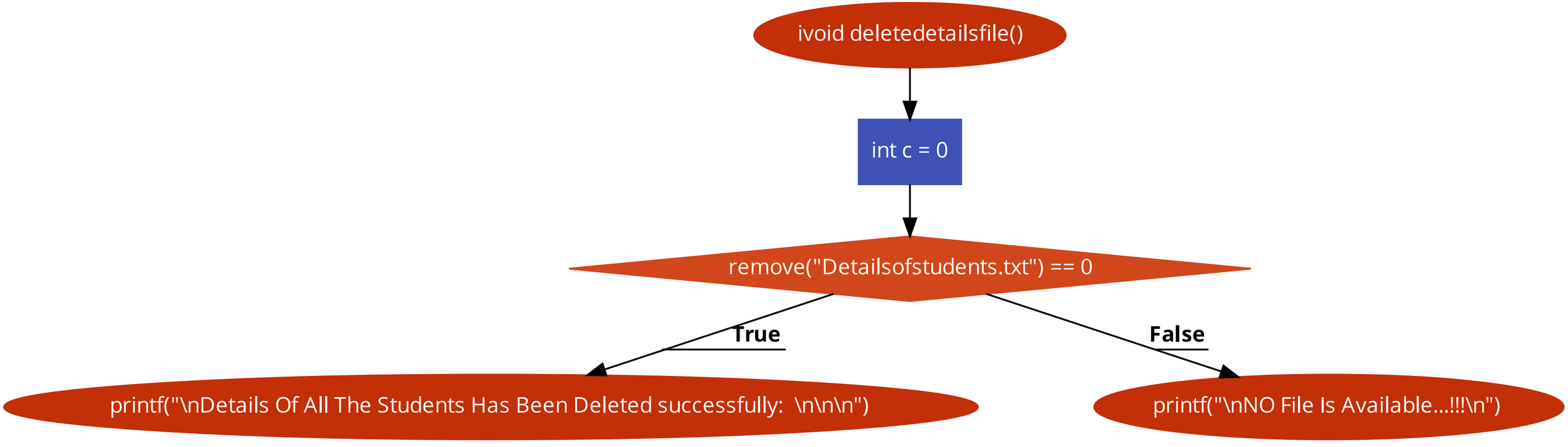

struct studentdetails
char sname
100
char fname
50
int cnic
int age
char adress
100
char gender
20
char rollnum
10
char section
20
int session
truct studentde
1


a = 0
a < b
True
False
printf("\nEnter Name Of The Student: ")
fflush(stdin)
gets(sd[a].sname)
printf("\nEnter Father's Name Of The Student: ")
fflush(stdin)
gets(sd[a].fname)
printf("\nEnter National Identity Card Number OF The Student Without (_) and (Empty Spacec): ")
fflush(stdin)
scanf("%d", &sd[a].cnic)
printf("\nEnter Age OF The Student: ")
fflush(stdin)
scanf("%d", &sd[a].age)
printf("\nGender OF The Student: ")
fflush(stdin)
scanf("%s", sd[a].gender)
printf("\nEnter Adress Of The Student: ")
fflush(stdin)
gets(sd[a].adress)
printf("\nEnter Roll_Number Of The Student: ")
fflush(stdin)
gets(sd[a].rollnum)
printf("\nEnter Section Of The Student: ")
fflush(stdin)
gets(sd[a].section)
printf("\nEnter Session Of The Student: ")
fflush(stdin)
scanf("%d", &sd[a].session)
printf("\n\n")
a++
fwrite(sd, sizeof(struct studentdetails), b, fp)
printf("Details of The students IS successfully Added...!\n\n\n")
fclose(fp)
void noofdetails() // FUNCTION DEFINITION TO CHECK NO STUDENTS ADDED IN THE DETAILS LIST.
struct studentdetails s1
FILE *fp*
fp = fopen("Detailsofstudents.txt", "r")
fseek(fp, 0, SEEK_END)
int n = ftell(fp) / sizeof(struct studentdetails)
printf("\n\nNo of Students Whose details are Saved = %d\n\n", n)
fclose(fp)

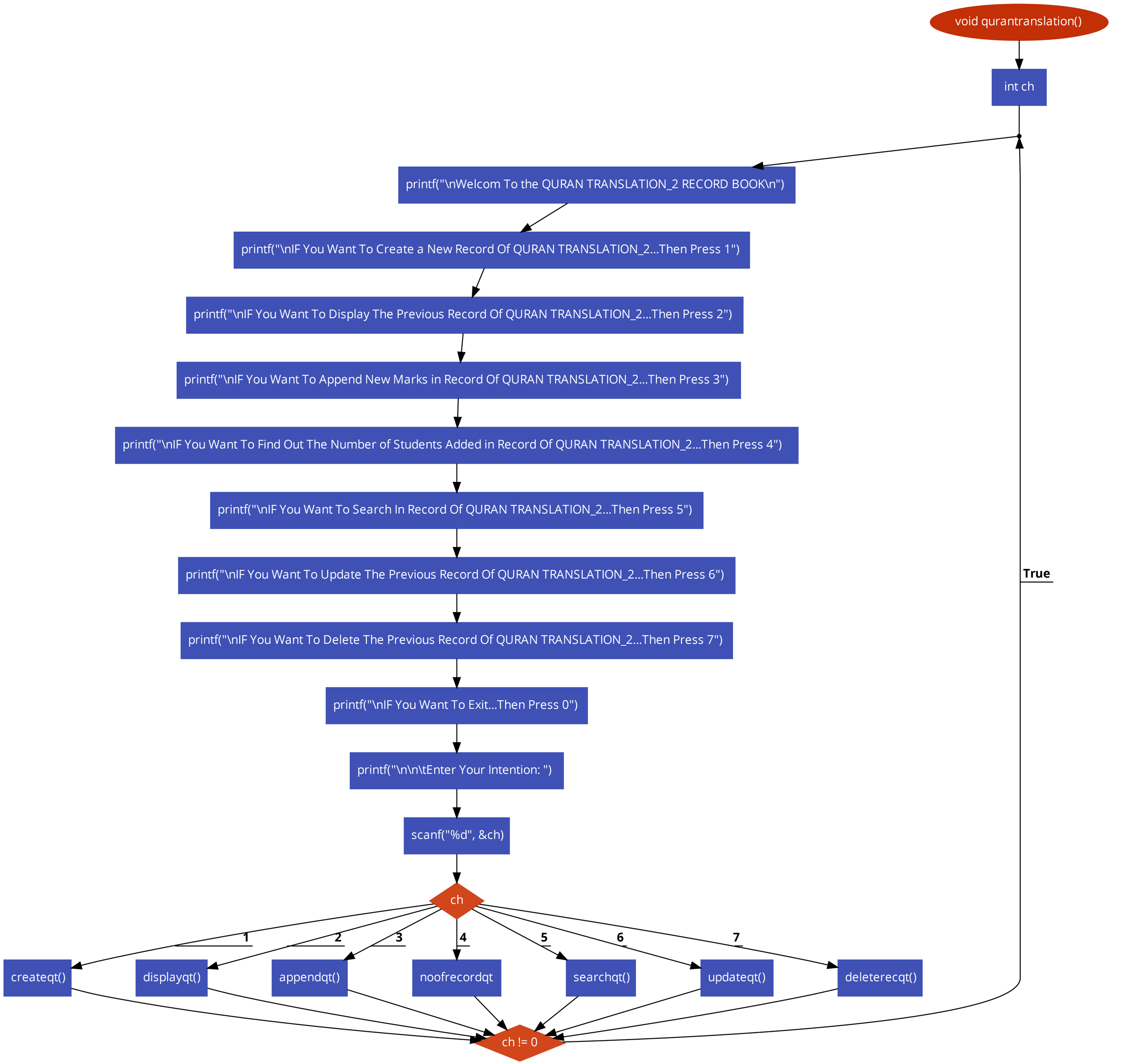

void addstudentdetails()
FILE *fp*
int a = 0
int b
int c = 500
printf("\nDetails of How many students do you want to add? ")
scanf("%d", &b)
b > c || b <= 0
True
printf("Invalid Entry...!!!")
printf("Try again...!!!")
printf("\nDetails of How many students do you want to add? ")
False
True
c <= b || b <= 0
False
struct studentdetails sd
500
printf("\nStart Adding The Details Of Students...!\n")
fp = fopen("Detailsofstudents.txt", "wb")
fp == NULL
True
printf("\nErrror in Opening the File...!!!")
exit(1)


void noofdetails()
struct studentdetails s1
FILE *fp*
fp = fopen("Detailsofstudents.txt", "r")
fseek(fp, 0, SEEK_END)
int n = ftell(fp) / sizeof(struct studentdetails)
printf("\n\nNo of Students Whose details are Saved = %d\n\n", n)
fclose(fp)

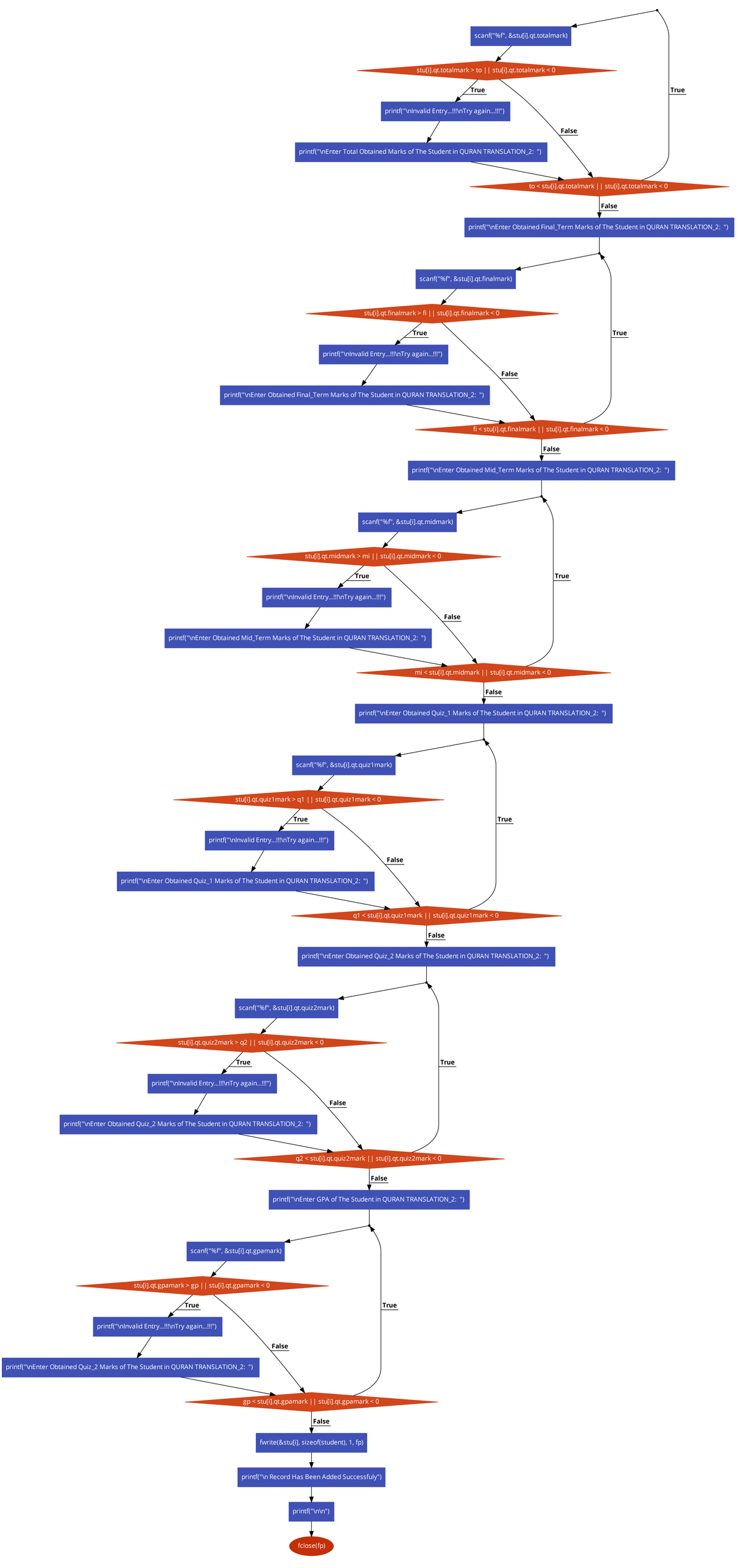

void removestudentdetails()
int j, found = 0
char rno
10
struct studentdetails s1
FILE *fp*, fp1
fp = fopen("Detailsofstudents.txt", "r")
fp1 = fopen("tempremove.txt", "w")
printf("\nEnter Roll_NO of the Student Whose Details you want To Delete: ")
scanf("%d", &rno)
printf("\n\n")
fread(&s1, sizeof(struct studentdetails), 1, fp)
False
True
fclose(fp)
strcmp(s1.rollnum, rno) == 0
False
True
fwrite(&s1, sizeof(struct studentdetails), 1, fp1)
found = 1
fclose(fp1)
found
True
False
fp1 = fopen("tempremove.txt", "r")
printf("\nDetails Of Student With this Roll_No are Not Found...!!!\n\n")
fp = fopen("Detailsofstudents.txt", "w")
fread(&s1, sizeof(struct studentdetails), 1, fp1)
True
False
fwrite(&s1, sizeof(struct studentdetails), 1, fp)
fclose(fp)
fclose(fp1)
printf("\nDetails Of This Students Have Been Removed Successfuly...!\n")
fclose(fp)


ivoid deletedetailsfile()
int c = 0
remove("Detailsofstudents.txt") == 0
True
False
printf("\nDetails Of All The Students Has Been Deleted successfully: \n\n\n")
printf("\nNO File Is Available...!!!\n")

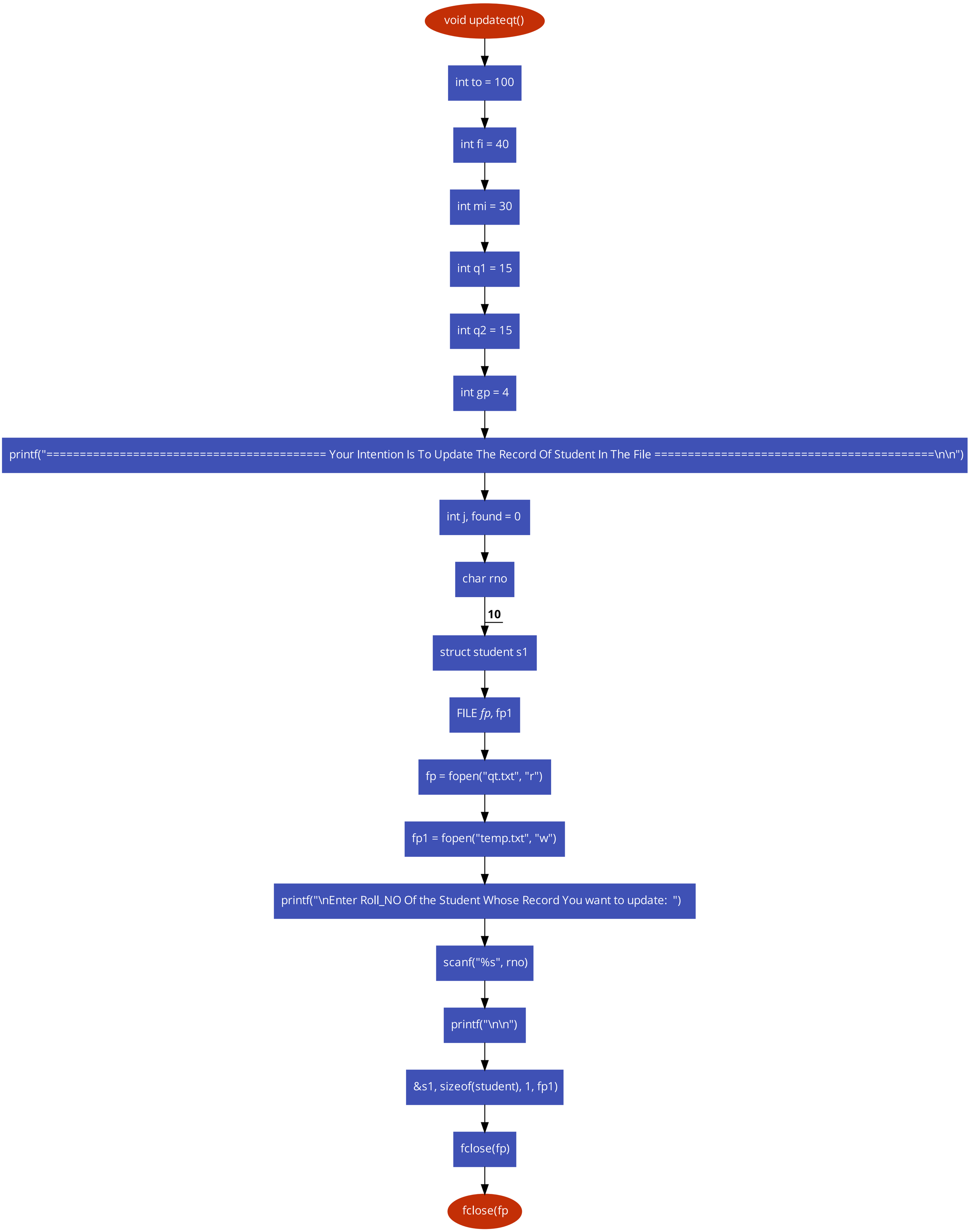
void opensubject()
char usercode
char vcacode[10] = "MA129"; // char cpcode[10] = "MCT242"; // Shar emcode[10] = "MCT223"; // SUar dldcode[10] = "MCT241"; // SUBr qtcode[10] = "QT201"; // SUBJE apcode[10] = "PHY118"; // SUBJEcagcode[10] = "MA113"; // SUBJECgdcode[10] = "MCT111"; // SUBJECT code[10] = "MCT121"; // SUBJECT code[10] = "ISL101"; // SUBJECT COEnter The Code Of Subject That You are Intending to Open. ");
10
char edcode[10] = "MCT211"
printf("\n\n\t\t\t__NOTE__\n\nSubject Code Should be In Capital Letters e.g:- MCT242...")
printf("\n\nEnter The Code Of Subject: ")
scanf("%s", &usercode)
strcmp(cpcode, usercode) == 0
True
computerprograming()
False
strcmp(emcode, usercode) == 0
True
electricmachinery()
False
strcmp(dldcode, usercode) == 0
True
digitallogicdesign()
False
strcmp(edcode, usercode) == 0
True
engineeringdynamics()
False
strcmp(vcacode, usercode) == 0
True
vectorandcoplexanalysis()
False
strcmp(qtcode, usercode) == 0
True
qurantranslation()
False
strcmp(egdcode, usercode) == 0
True
graphicsanddesign()
False
strcmp(eccode, usercode) == 0
True
electriccircuits()
False
strcmp(apcode, usercode) == 0
True
appliapphysics()
False
strcmp(cagcode, usercode) == 0
True
calculas()
False
strcmp(islcode, usercode) == 0
True
islamicandpakstudies()

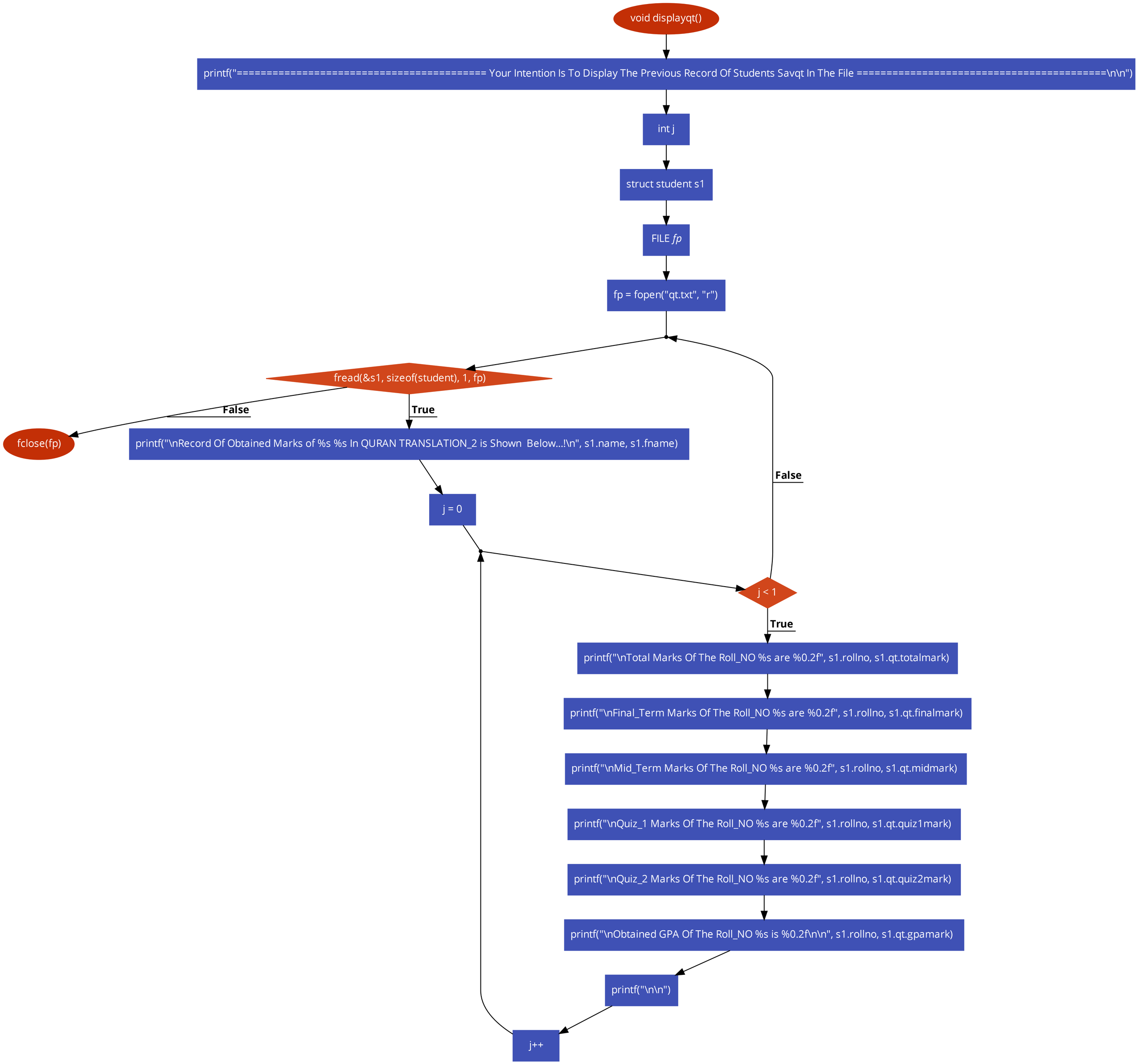

void qurantranslation()
int ch
printf("\nWelcom To the QURAN TRANSLATION_2 RECORD BOOK\n")
printf("\nIF You Want To Create a New Record Of QURAN TRANSLATION_2...Then Press 1")
printf("\nIF You Want To Display The Previous Record Of QURAN TRANSLATION_2...Then Press 2")
printf("\nIF You Want To Append New Marks in Record Of QURAN TRANSLATION_2...Then Press 3")
printf("\nIF You Want To Find Out The Number of Students Added in Record Of QURAN TRANSLATION_2...Then Press 4")
printf("\nIF You Want To Search In Record Of QURAN TRANSLATION_2...Then Press 5")
printf("\nIF You Want To Update The Previous Record Of QURAN TRANSLATION_2...Then Press 6")
printf("\nIF You Want To Delete The Previous Record Of QURAN TRANSLATION_2...Then Press 7")
printf("\nIF You Want To Exit...Then Press 0")
printf("\n\n\tEnter Your Intention: ")
scanf("%d", &ch)
ch
1
2
3
4
5
6
7
createqt()
displayqt()
appendqt()
noofrecordqt
searchqt()
updateqt()
deleterecqt()
ch != 0
True

void createqt()
printf("========== Your Intention Is To Add The Record Of Students In The File ==========\n\n")
struct student *stu*
FILE *fp*
int n, i, j
int k = 500
int to = 100
int fi = 40
int mi = 30
int q1 = 15
int q2 = 15
int gp = 4
printf("\nResult of How Many Students do you want to add? ")
scanf("%d", &n)
n > k || n < 0
True
printf("\nInvalid Entry...!!!")
printf("\nTry again...!!!")
printf("\nResult of How Many Students do you want to add? ")
False
True
k <= n || n <= 0
False
printf("\n\nStart Adding The Result Of The Students...!\n")
printf("\n_____NOTE_____\n\nTotal Marks Of Subject are 100\nTotal Marks Of Final_Term are 40\nTotal Marks Of Mid_Term are 40\nQuiz_1 Marks are 15\nQuiz_2 Marks are 15\nMaximum Grade Point Average Is 4\n\n")
stu = (struct student *)calloc(n, sizeof(student))
fp = fopen("qt.txt", "w")
i = 0
i < n
True
printf("\nEnter Name Of The Student: ")
scanf("%s %s", &stu[i].name, &stu[i].fname)
fflush(stdin)
printf("\nEnter Roll No Of The Student: ")
scanf("%s", &stu[i].rollno)
printf("\nEnter Total Obtained Marks of The Student in QURAN TRANSLATION_2: ")
i++

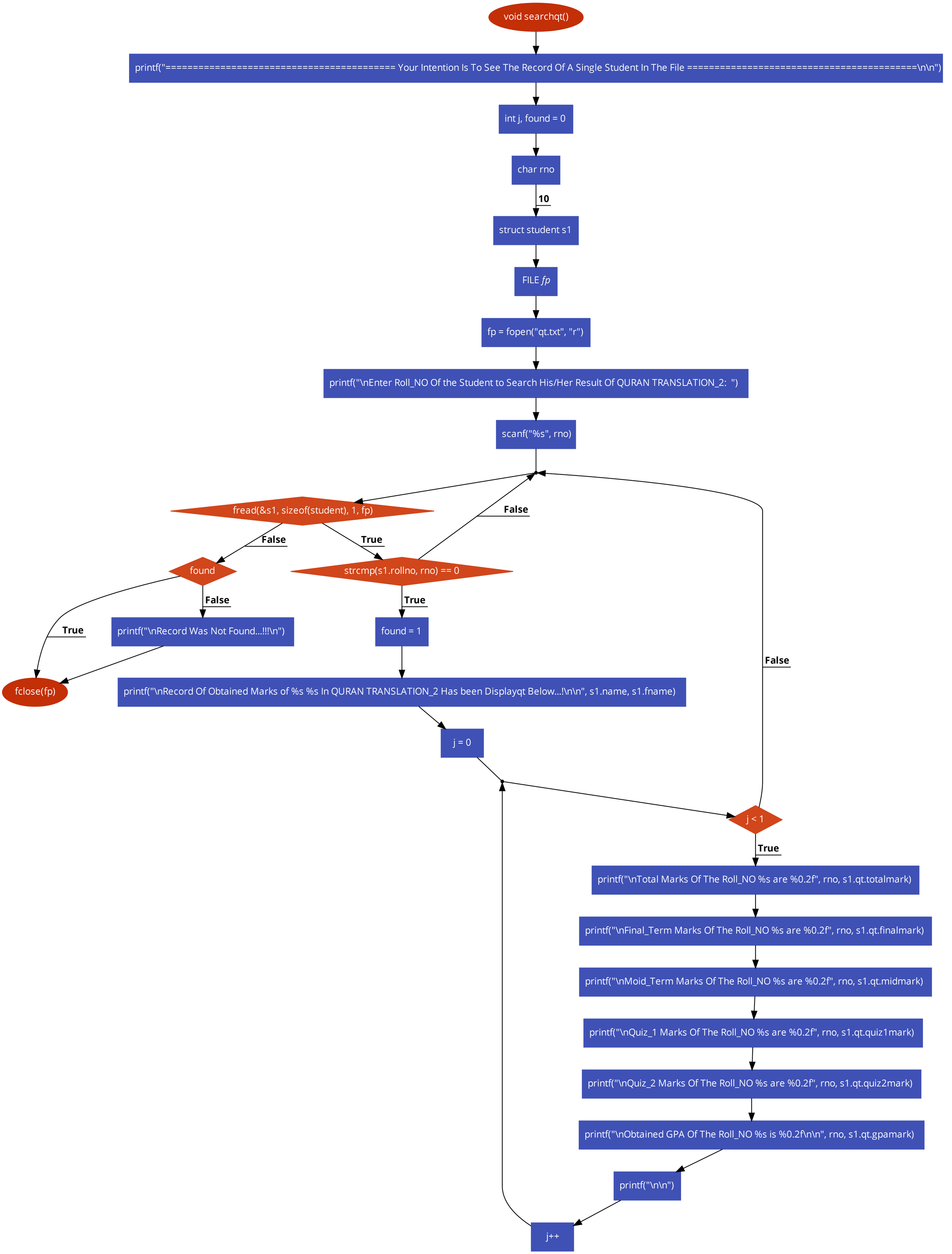

scanf("%f", &stu[i].qt.totalmark)
stu[i].qt.totalmark > to || stu[i].qt.totalmark < 0
True
False
printf("\nInvalid Entry...!!!\nTry again...!!!")
printf("\nEnter Total Obtained Marks of The Student in QURAN TRANSLATION_2: ")
to < stu[i].qt.totalmark || stu[i].qt.totalmark < 0
printf("\nEnter Obtained Final_Term Marks of The Student in QURAN TRANSLATION_2: ")
scanf("%f", &stu[i].qt.finalmark)
stu[i].qt.finalmark > fi || stu[i].qt.finalmark < 0
printf("\nInvalid Entry...!!!\nTry again...!!!")
printf("\nEnter Obtained Final_Term Marks of The Student in QURAN TRANSLATION_2: ")
fi < stu[i].qt.finalmark || stu[i].qt.finalmark < 0
printf("\nEnter Obtained Mid_Term Marks of The Student in QURAN TRANSLATION_2: ")
scanf("%f", &stu[i].qt.midmark)
stu[i].qt.midmark > mi || stu[i].qt.midmark < 0
printf("\nInvalid Entry...!!!\nTry again...!!!")
printf("\nEnter Obtained Mid_Term Marks of The Student in QURAN TRANSLATION_2: ")
mi < stu[i].qt.midmark || stu[i].qt.midmark < 0
printf("\nEnter Obtained Quiz_1 Marks of The Student in QURAN TRANSLATION_2: ")
scanf("%f", &stu[i].qt.quiz1mark)
stu[i].qt.quiz1mark > q1 || stu[i].qt.quiz1mark < 0
printf("\nInvalid Entry...!!!\nTry again...!!!")
printf("\nEnter Obtained Quiz_1 Marks of The Student in QURAN TRANSLATION_2: ")
q1 < stu[i].qt.quiz1mark || stu[i].qt.quiz1mark < 0
printf("\nEnter Obtained Quiz_2 Marks of The Student in QURAN TRANSLATION_2: ")
scanf("%f", &stu[i].qt.quiz2mark)
stu[i].qt.quiz2mark > q2 || stu[i].qt.quiz2mark < 0
printf("\nInvalid Entry...!!!\nTry again...!!!")
printf("\nEnter Obtained Quiz_2 Marks of The Student in QURAN TRANSLATION_2: ")
q2 < stu[i].qt.quiz2mark || stu[i].qt.quiz2mark < 0
printf("\nEnter GPA of The Student in QURAN TRANSLATION_2: ")
scanf("%f", &stu[i].qt.gpamark)
stu[i].qt.gpamark > gp || stu[i].qt.gpamark < 0
printf("\nInvalid Entry...!!!\nTry again...!!!")
printf("\nEnter Obtained Quiz_2 Marks of The Student in QURAN TRANSLATION_2: ")
gp < stu[i].qt.gpamark || stu[i].qt.gpamark < 0
fwrite(&stu[i], sizeof(student), 1, fp)
printf("\n Record Has Been Added Successfuly")
printf("\n\n")
fclose(fp)

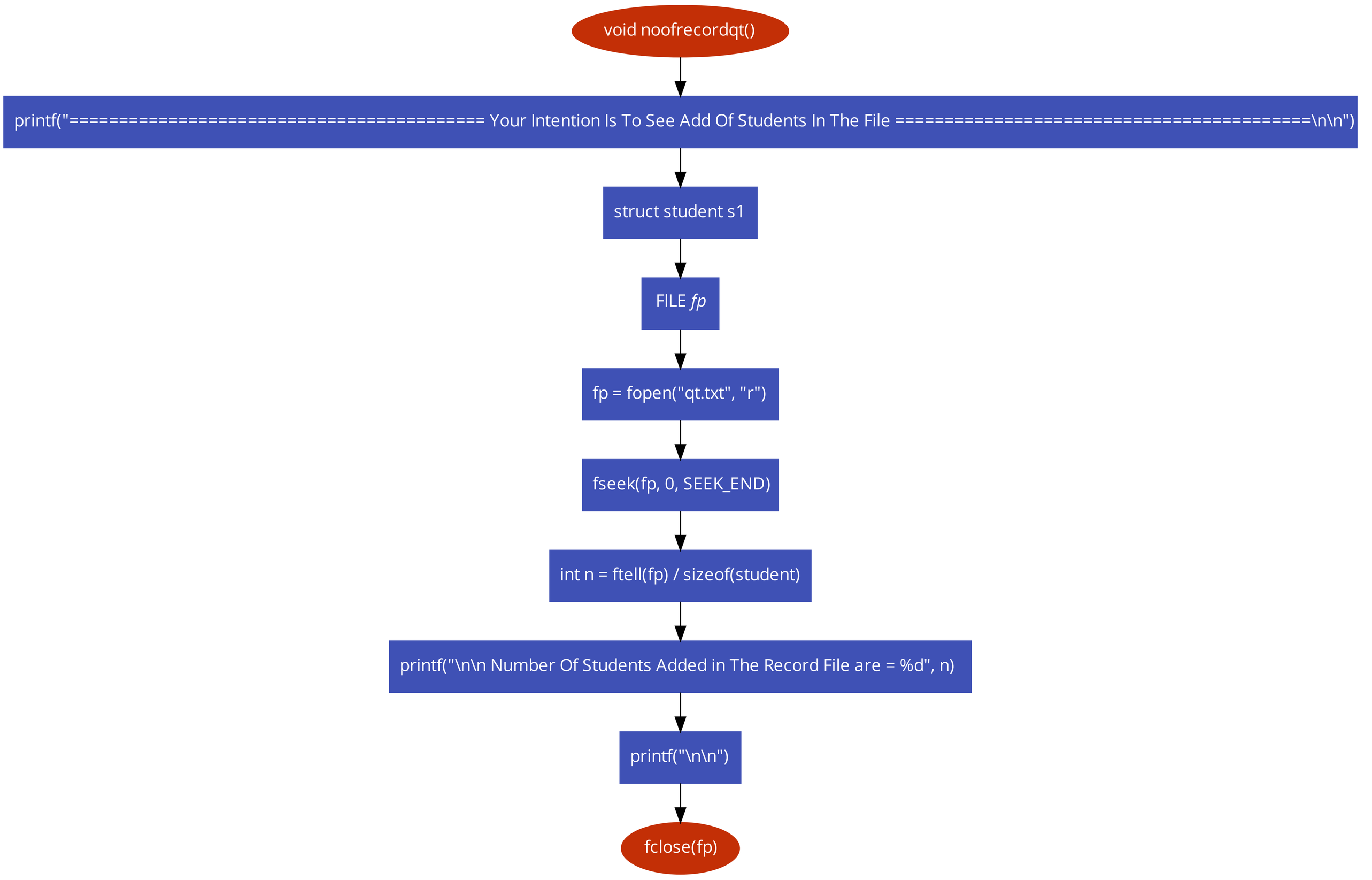
void displaydetails()
int j
struct studentdetails s1
FILE *fp*
fp = fopen("Detailsofstudents.txt", "r")
fread(&s1, sizeof(struct studentdetails), 1, fp)
True
False
fclose(fp)
printf("\nSaved Details OF %s is Shown Below...!\n", s1.sname)
printf("\nFather Name Of %s Is %s", s1.sname, s1.fname)
printf("\nCNIC Of %s Is %d", s1.sname, s1.cnic)
printf("\nAge Of %s Is %d", s1.sname, s1.age)
printf("\nGender Of student is %s", s1.gender)
printf("\nAdress Of %s Is %s", s1.sname, s1.adress)
printf("\nRoll_No Of %s Is %s", s1.sname, s1.rollnum)
printf("\nSection Of %s Is %s", s1.sname, s1.section)
printf("\nSession Of %s Is %d", s1.sname, s1.session)
printf("\n\n")

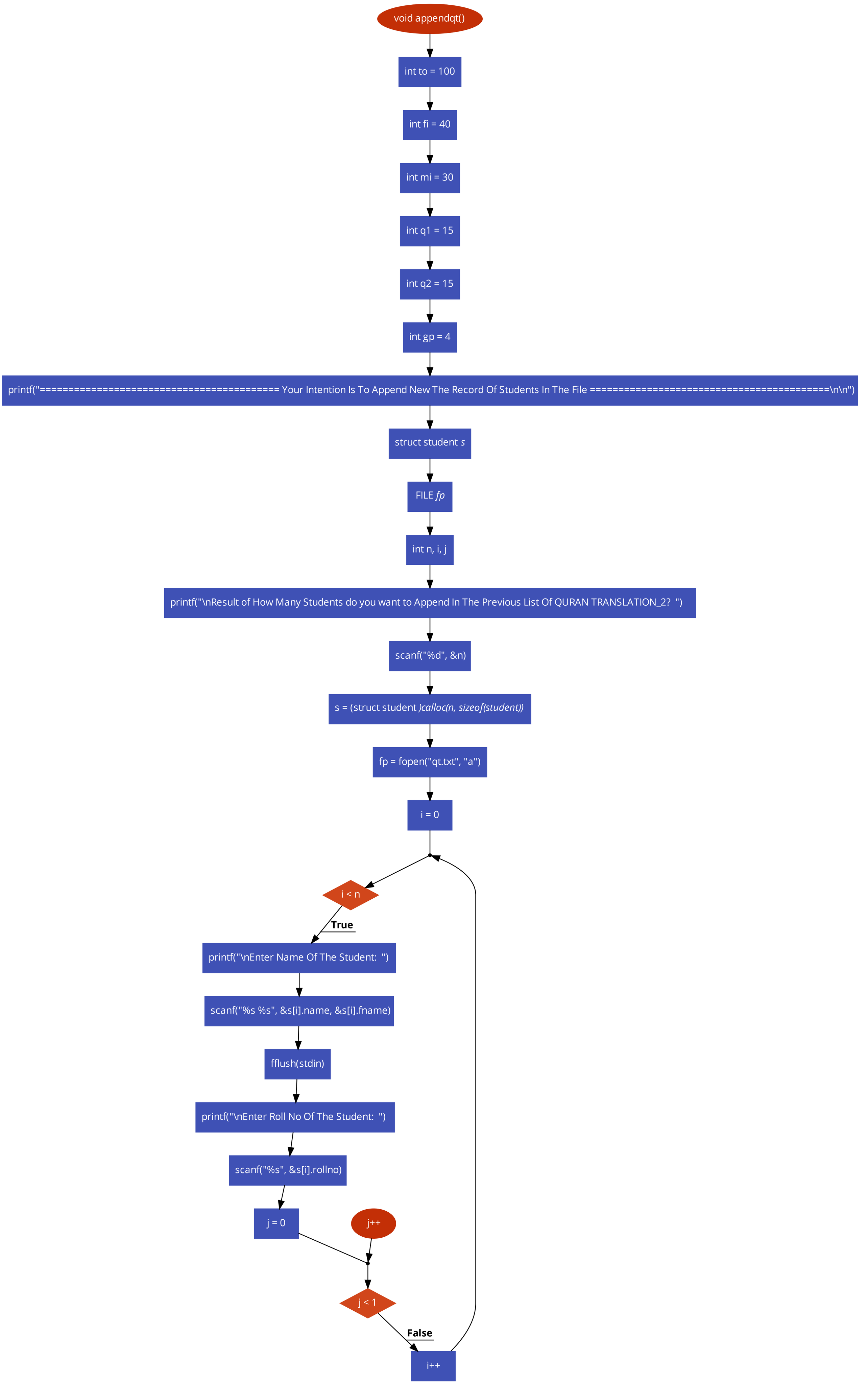

void updateqt()
int to = 100
int fi = 40
int mi = 30
int q1 = 15
int q2 = 15
int gp = 4
printf("== Your Intention Is To Update The Record Of Student In The File ==\n\n")
int j, found = 0
char rno
10
struct student s1
FILE fp, fp1
fp = fopen("qt.txt", "r")
fp1 = fopen("temp.txt", "w")
printf("\nEnter Roll_NO Of the Student Whose Record You want to update: ")
scanf("%s", rno)
printf("\n\n")
&s1, sizeof(student), 1, fp1)
fclose(fp)
fclose(fp

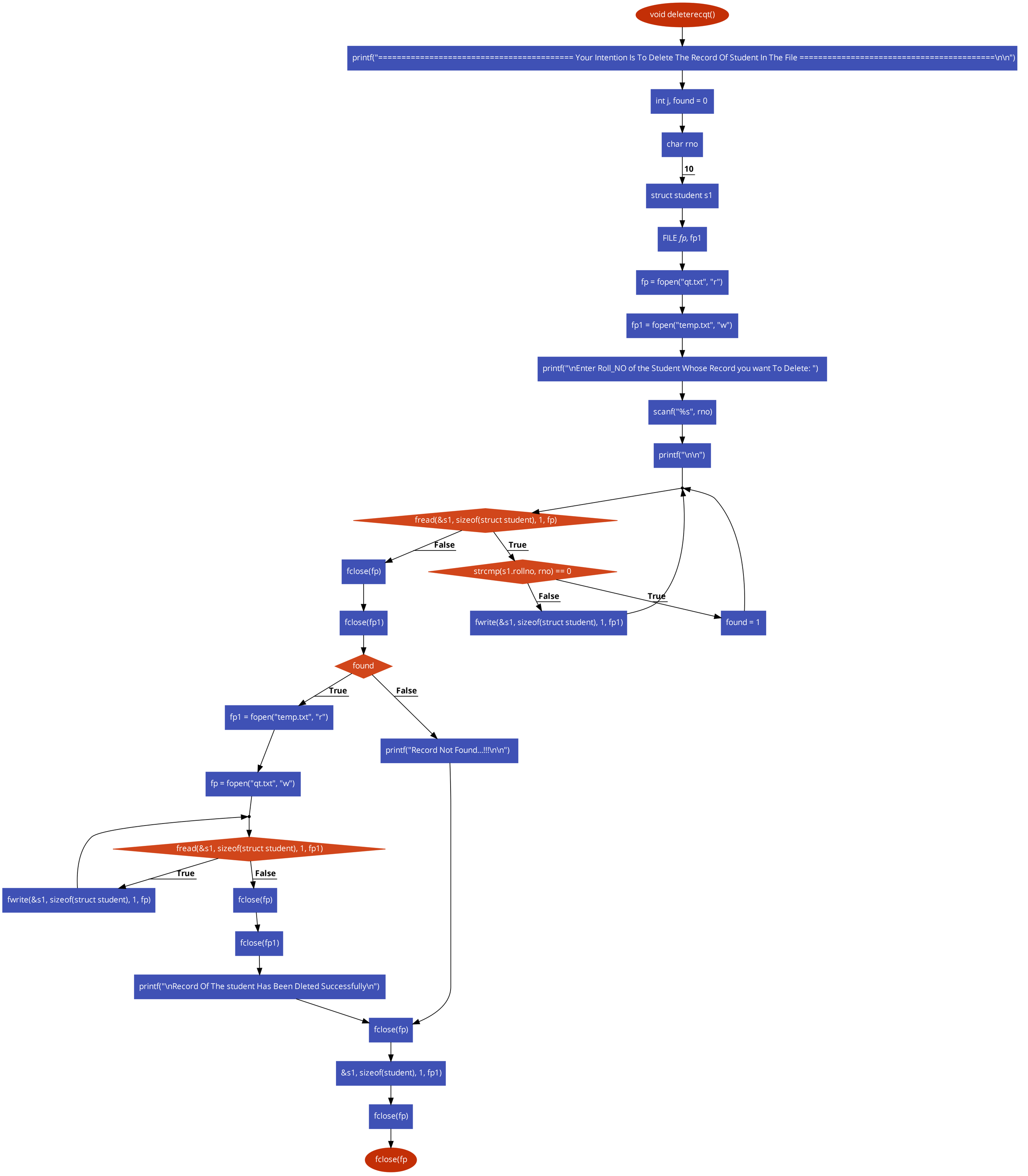

void displayqt()
printf("========== Your Intention Is To Display The Previous Record Of Students Savqt In The File ==========\n\n")
int j
struct student s1
FILE *fp*
fp = fopen("qt.txt", "r")
fread(&s1, sizeof(student), 1, fp)
False
True
fclose(fp)
printf("\nRecord Of Obtained Marks of %s %s In QURAN TRANSLATION_2 is Shown Below...!\n", s1.name, s1.fname)
j = 0
j < 1
False
True
printf("\nTotal Marks Of The Roll_NO %s are %0.2f", s1.rollno, s1.qt.totalmark)
printf("\nFinal_Term Marks Of The Roll_NO %s are %0.2f", s1.rollno, s1.qt.finalmark)
printf("\nMid_Term Marks Of The Roll_NO %s are %0.2f", s1.rollno, s1.qt.midmark)
printf("\nQuiz_1 Marks Of The Roll_NO %s are %0.2f", s1.rollno, s1.qt.quiz1mark)
printf("\nQuiz_2 Marks Of The Roll_NO %s are %0.2f", s1.rollno, s1.qt.quiz2mark)
printf("\nObtained GPA Of The Roll_NO %s is %0.2f\n\n", s1.rollno, s1.qt.gpamark)
printf("\n\n")
j++

fwrite(&s1, sizeof(student), 1, fp1)
fclose(fp)
fclose(fp1)
found
True
False
fp1 = fopen("temp.txt", "r")
printf("Error,Record Not Found...!!!\n\n")
fp = fopen("qt.txt", "w")
&s1, sizeof(student), 1, fp1)
fread(&s1, sizeof(student), 1, fp1)
True
False
fwrite(&s1, sizeof(student), 1, fp)
fclose(fp)
fclose(fp1)
printf("\nRecord of the Student %s Has Been Updated Successfully...!!!", s1.rollno)
printf("\n\n")
fclose(fp)
fclose(fp


void searchqt()
printf("========== Your Intention Is To See The Record Of A Single Student In The File ==========\n\n")
int j, found = 0
char rno
10
struct student s1
FILE *fp
fp = fopen("qt.txt", "r")
printf("\nEnter Roll_NO Of the Student to Search His/Her Result Of QURAN TRANSLATION_2: ")
scanf("%s", rno)
fread(&s1, sizeof(student), 1, fp)
False
True
found
strcmp(s1.rollno, rno) == 0
False
True
True
False
printf("\nRecord Was Not Found...!!!\n")
found = 1
fclose(fp)
printf("\nRecord Of Obtained Marks of %s %s In QURAN TRANSLATION_2 Has been Displayqt Below...!\n\n", s1.name, s1.fname)
j = 0
j < 1
True
False
printf("\nTotal Marks Of The Roll_NO %s are %0.2f", rno, s1.qt.totalmark)
printf("\nFinal_Term Marks Of The Roll_NO %s are %0.2f", rno, s1.qt.finalmark)
printf("\nMoid_Term Marks Of The Roll_NO %s are %0.2f", rno, s1.qt.midmark)
printf("\nQuiz_1 Marks Of The Roll_NO %s are %0.2f", rno, s1.qt.quiz1mark)
printf("\nQuiz_2 Marks Of The Roll_NO %s are %0.2f", rno, s1.qt.quiz2mark)
printf("\nObtained GPA Of The Roll_NO %s is %0.2f\n\n", rno, s1.qt.gpamark)
printf("\n\n")
j++


void noofrecordqt()
printf("========== Your Intention Is To See Add Of Students In The File ==========\n\n")
struct student s1
FILE *fp*
fp = fopen("qt.txt", "r")
fseek(fp, 0, SEEK_END)
int n = ftell(fp) / sizeof(student)
printf("\n\n Number Of Students Added in The Record File are = %d", n)
printf("\n\n")
fclose(fp)


void appendqt()
int to = 100
int fi = 40
int mi = 30
int q1 = 15
int q2 = 15
int gp = 4
printf("======== Your Intention Is To Append New The Record Of Students In The File ========\n\n")
struct student *s*
FILE *fp*
int n, i, j
printf("\nResult of How Many Students do you want to Append In The Previous List Of QURAN TRANSLATION_2? ")
scanf("%d", &n)
s = (struct student *)calloc(n, sizeof(student))
fp = fopen("qt.txt", "a")
i = 0
i < n
True
printf("\nEnter Name Of The Student: ")
scanf("%s %s", &s[i].name, &s[i].fname)
fflush(stdin)
printf("\nEnter Roll No Of The Student: ")
scanf("%s", &s[i].rollno)
j = 0
j++
j < 1
False
i++


void deleterecqt()
printf("========== Your Intention Is To Delete The Record Of Student In The File ==========\n\n")
int j, found = 0
char rno
10
struct student s1
FILE *fp*, fp1
fp = fopen("qt.txt", "r")
fp1 = fopen("temp.txt", "w")
printf("\nEnter Roll_NO of the Student Whose Record you want To Delete: ")
scanf("%s", rno)
printf("\n\n")
fread(&s1, sizeof(struct student), 1, fp)
False
True
fclose(fp)
strcmp(s1.rollno, rno) == 0
False
True
fclose(fp1)
fwrite(&s1, sizeof(struct student), 1, fp1)
found = 1
found
True
False
fp1 = fopen("temp.txt", "r")
printf("Record Not Found...!!!\n\n")
fp = fopen("qt.txt", "w")
fread(&s1, sizeof(struct student), 1, fp1)
True
False
fwrite(&s1, sizeof(struct student), 1, fp)
fclose(fp)
fclose(fp1)
printf("\nRecord Of The student Has Been Dleted Successfully\n")
fclose(fp)
&s1, sizeof(student), 1, fp1)
fclose(fp)
fclose(fp