**Function 1:-**

**Void load(void); //Loading Screen Function:-**

This function is called from the main function in start just used ti enter loading when user starts the game. First it should change the output screen color then print the given things in code and then show the loading on the output screen

**Source code:-**

void load(void)

{

int i;

system("color 6");

system("cls");

gotoxy(70,5);

printf("CROSSWORD GAME");

gotoxy(70,10);

printf("LOADING");

gotoxy(70,15);

printf("GROUP No 1");

gotoxy(77,10);

for(i=0;i<4;i++)

{

Sleep(200);

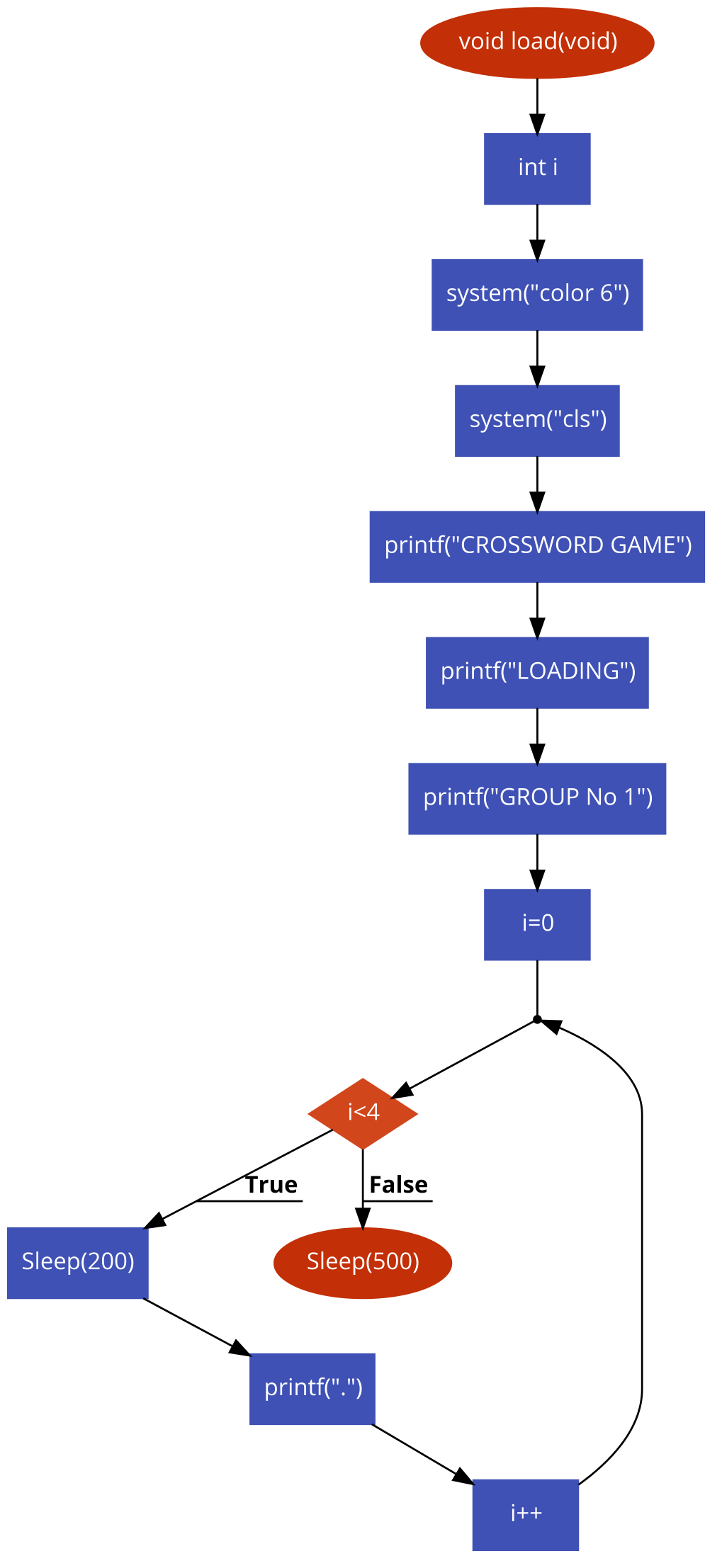
printf(".");

}

Sleep(500);

}

**Flowchart:-**



**Function 2:-**

**void gotoxy (int x, int y); //Function for changing the cursor position:-**

This function is used just to change the position of x and y onto the output screen to display what is written in it. It is basically changing the cursor position.

**Source code:-**

//Function for changing the cursor position

void gotoxy (int x, int y)

{

COORD coord = {0, 0};

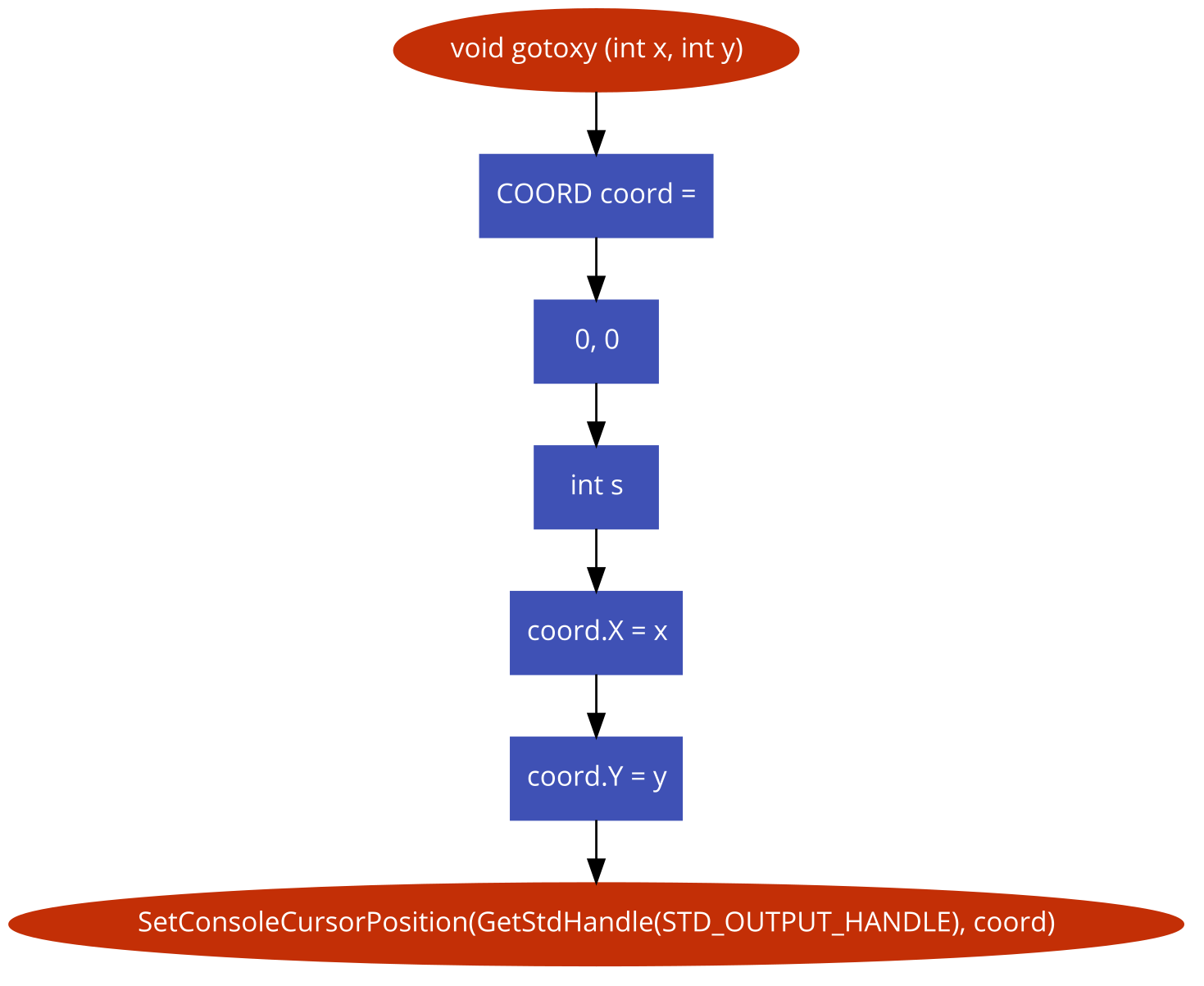
int s;

coord.X = x; coord.Y = y;

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), coord);

}

**Flowchart:-**



**Function 3:-**

**void startgame (char username[],int life,int level); // Function to start a game:-**

Function is called from the main function and is used to for the level slection from the user. Half of its work done in main means printing or else condition used in main function. From this function user enter the level if level is from 1,2,3 than program go forward other wise else condition is used and goto funtction is used and used until the correct level is added by user.

**Source code:-**

int levelslect(int level)

{

scanf("%d",&level);

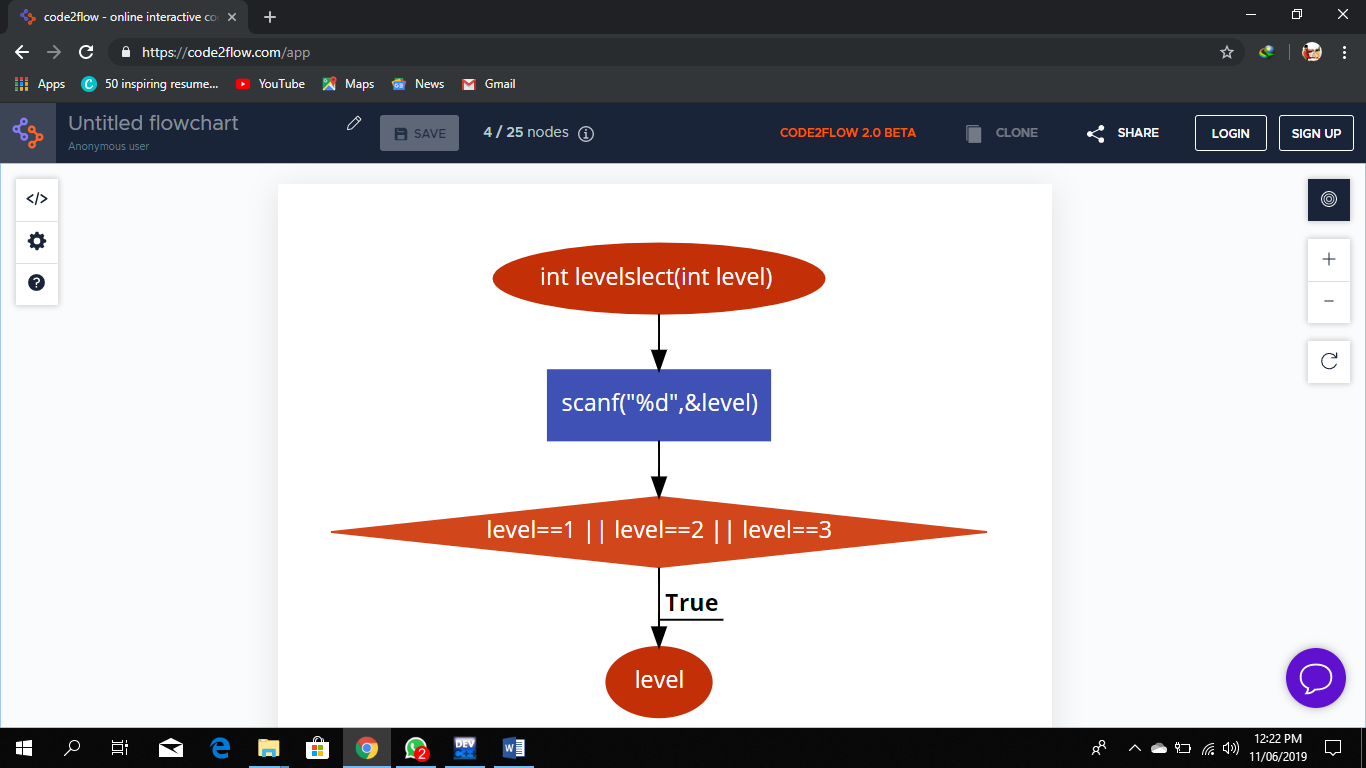
if (level==1 || level==2 || level==3)

{ return level;

}

}

**Flowchart:-**



**Function 4:-**

**int newgame(int level); //Function to get the username of a player and store it in file:-**

After entering the level function the code moves forward and the funvtion of new game is called from the main function which enter the user to new game every time it starts. New game will store the data of user name or level etc which are used for a new game everytime. And than call the new function startgame.

**Source code:-**

int newgame(int level)

{ system("color A");

system("color B");

system("CLS");

struct ng a;

FILE \*fp;

fp=fopen("Username.dat","a+");

gotoxy(70,1);

printf("Enter Username:");

scanf("%s",a.username);

fprintf(fp,"%s\n",a.username);

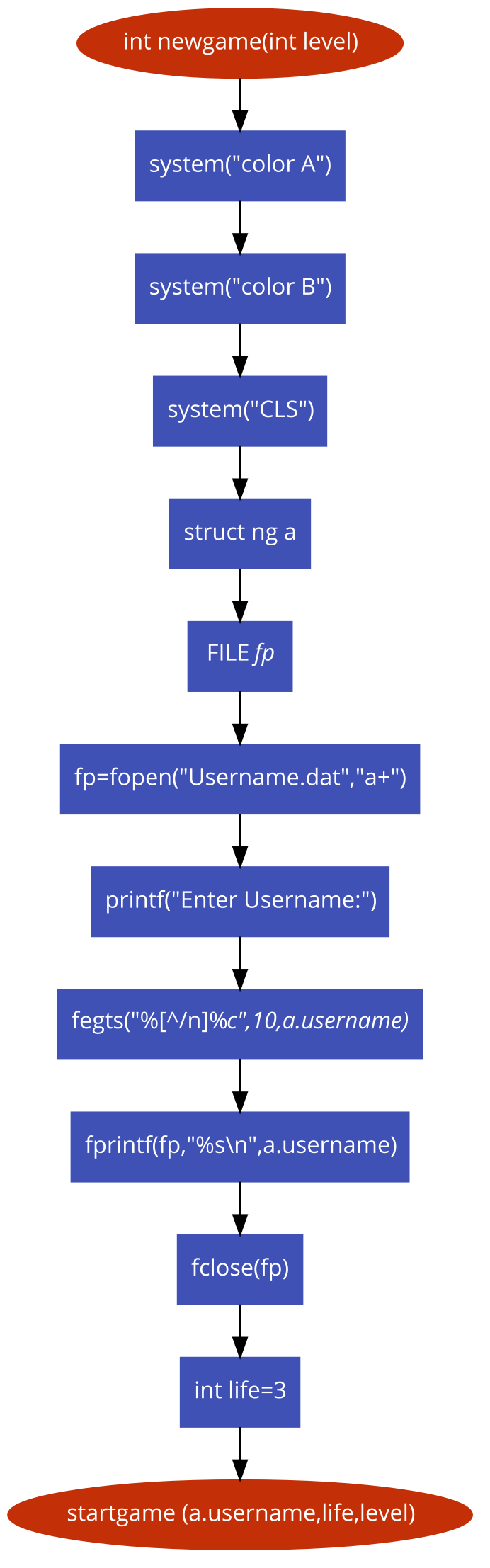
fclose(fp);

int life=3;

startgame (a.username,life,level);

}

**Flowchart:-**



**Function 5:-**

**void startgame (char username[],int life,int level); // Function to start a game:-**

As this function is called from the previous function and returning no values. After user enter the data for new game this function is called using the given data and pass values to new function to start the game. It works according the given level and which level is given according to that the if condition runs and calls the new function to proceed the code. After the given if condition first it calls the function to read and then function of board to read board.

**Source code: -**

void startgame (char username[],int life,int level)

{

system("color 50");//this function is use to change the screen color

system("color e7");//this is also used to change the screen color along with previous one on running time

char input[32],t;

system("CLS");

printf("Name of player is %s\n",username);

if(level==1)

{

int l=7,o=7,score=0;

printf("your selected level is 1\n");

read(level);

position(l,o,level,life,score);

}

else if(level==2)

{int l=10,o=10,score=0;;

read(level);

position(l,o,level,life,score);

}

else if(level==3)

{ int l=12,o=7,score=0;

printf("Your slected level is 3\n");

read(level);

position(l,o,level,life,score);

}

/\*if (life<=0);

{

printf("Game over");

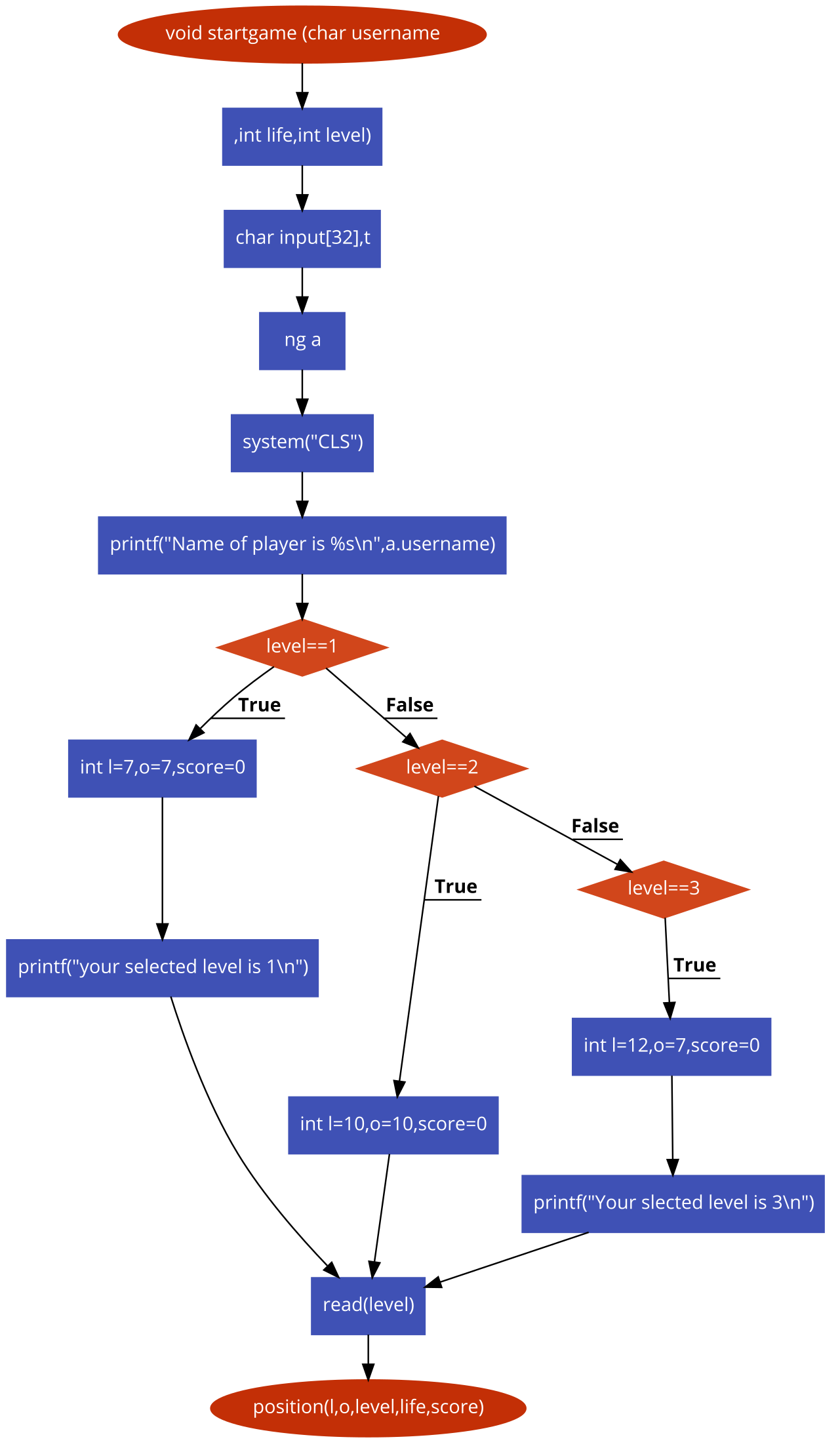
getch();

break;

}\*/

}

**Flow chart: -**



**Function 6:-**

**char read(int level); //Funtion contains the words to be search in the give board:-**

This is the function which is used to give the previous given data and show it on output screen i.e. user name etc. but before this it called another function to read board than proceed and than it is given the words to find according to the given level. Here gotoxy also used to change the position of recursor and if conditions are working according to the given level which we take in start. This is char type function means that it returns only characters. it calls the function of read board according to specific slection statement

**Source code:-**

char read(int level)

{

if(level==1)

{

readBoard();

gotoxy(0,27);

printf("Easy Level\n");

gotoxy(68,10);

printf("Your Given Words are\n");

gotoxy(70,11);

printf("1-free"); gotoxy(70,12);

printf("2-comsats");gotoxy(70,13);

printf("3-app");gotoxy(70,14);

printf("4-boy");gotoxy(70,15);

printf("5-car");gotoxy(70,16);

printf("6-camera");gotoxy(70,17);

printf("7-pack");gotoxy(70,18);

printf("8-gas");gotoxy(70,19);

}

else if(level==2)

{

readBoard2();

gotoxy(0,27);

printf("Medium Level\n");

gotoxy(68,10);

printf("Your Given Words are\n");

gotoxy(70,11);

printf("1-Apple");gotoxy(70,12);

printf("2-mango");gotoxy(70,13);

printf("3-bnana");gotoxy(70,14);

printf("4-cooler");gotoxy(70,15);

printf("5-gas");gotoxy(70,16);

printf("6-oppo");gotoxy(70,17);

printf("7-mobile");gotoxy(70,18);

}

else if(level==3)

{

readBoard3();

gotoxy(0,27);

printf("Hard Level\n");

gotoxy(68,10);

printf("Your Given Words are\n");

gotoxy(70,11);

printf("1-camera");gotoxy(70,12);

printf("2-apple");gotoxy(70,13);

printf("3-boy");gotoxy(70,14);

printf("4-mango");gotoxy(70,15);

printf("5-gas");gotoxy(70,16);

printf("6-oppo");gotoxy(70,17);

printf("7-mobile");gotoxy(70,18);

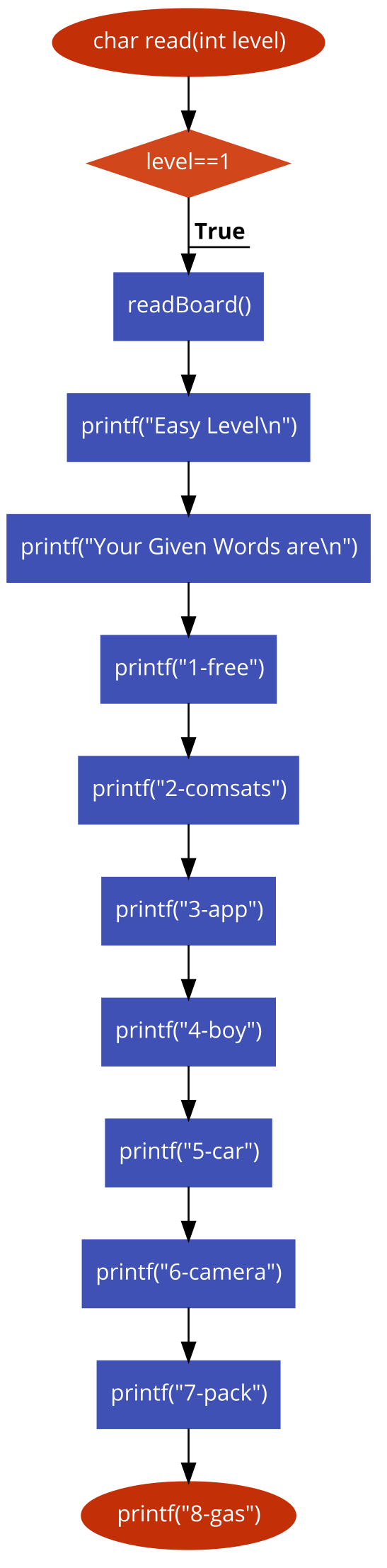
printf("8-comsats");gotoxy(70,19);

printf("9-oil");gotoxy(70,20);

printf("10-bulb");gotoxy(70,21);

}

**Flow chart:-**



The flowchart for read2(level); and read3(level); is same as above.

**Function 7,8,9:-**

**Here we use all three boards of selection statement**

**char readBoard(); //Fuction to open baord text file and print it on screen:-**

**char readBoard2(); //Fuction to open baord text file and print it on screen:-**

**char readBoard3(); //Fuction to open baord text file and print it on screen:-**

This function is first used in the start of the previous function to read the board from where user has to find the word first this board is shown in the screen because it terminated first and than the previous function runs. This is alse selected on the basis of level selected before.

**Source code:-**

char readBoard()

{

char ch;

FILE \*fp;

fp = fopen("board.txt", "r");

while(1)

{

ch=fgetc (fp);

if (ch==EOF)

break;

printf("%c",ch);

}

fclose(fp);

return 0;

}

//Fuction to open baord text file and print it on screen

char readBoard2()

{

char ch;

FILE \*fp;

fp = fopen("board2.txt", "r");

while(1)

{

ch=fgetc (fp);

if (ch==EOF)

break;

printf("%c",ch);

}

fclose(fp);

return 0;

}

//Fuction to open baord text file and print it on screen

char readBoard3()

{

char ch;

FILE \*fp;

fp = fopen("board3.txt", "r");

while(1)

{ch=fgetc (fp);

if (ch==EOF)

break;

printf("%c",ch);

}

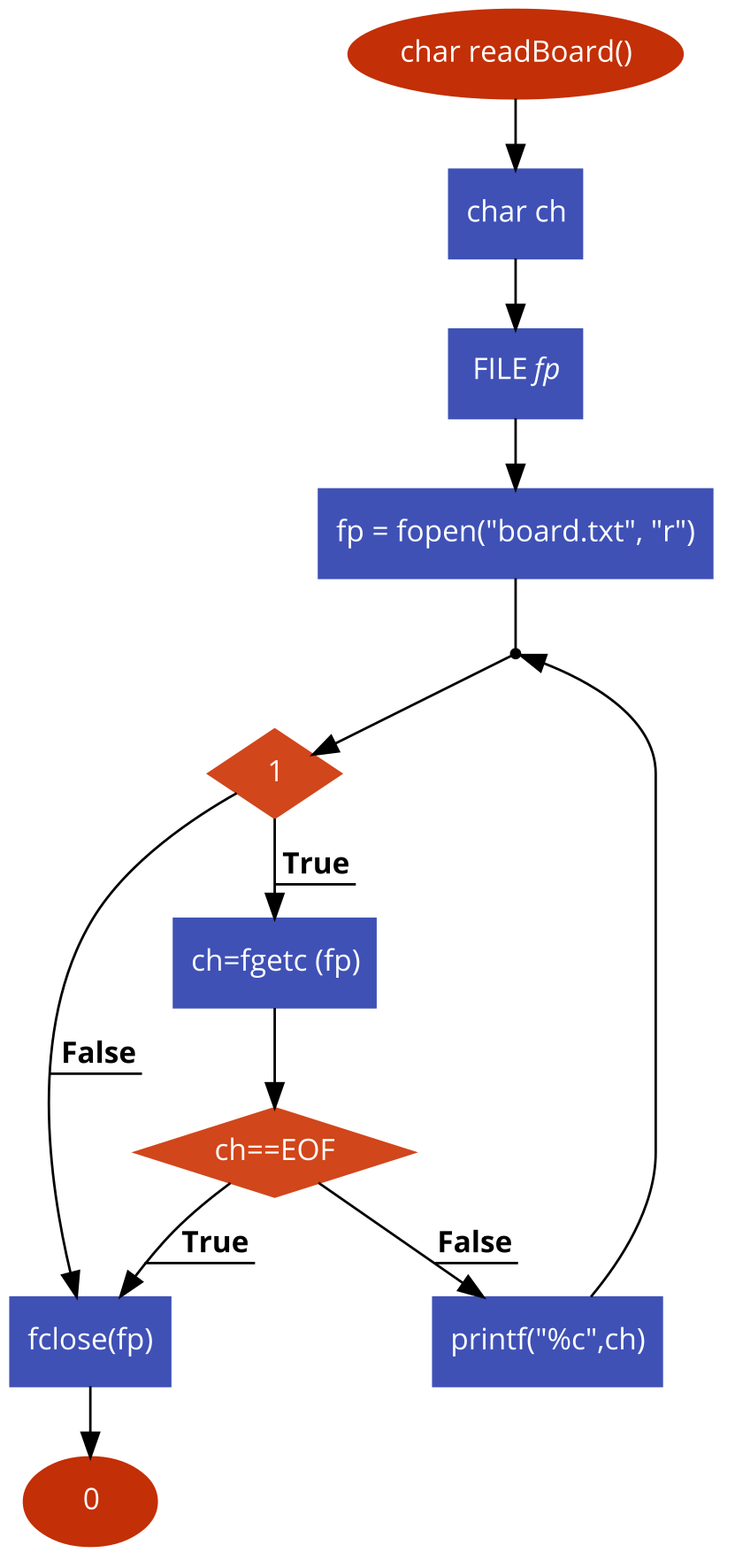
fclose(fp);

return 0;

}

**Flow chart:-**

We make the flow chart of only one board because the logic is same for readboard2 and readboard3:-



**Function 10:-**

**int position(int l , int o, int level,int life, int score); //Fuction to select the position of seraching within board:-**

After reading the board and read all the given data and print it on the output screen the code flows back to the function of **startgame** and here it used the next function named as position. Which asked from the user that from which position he/she want to find the word i.e. forward bise? , reverse bise? Or etc. From this function the furture code exeeds first it find the finding position from user and make it equal to the n and than using the n for exeed in game by usinf selection statement. It is basically given 8 searching positions.

**Source code:-**

int position(int l, int o, int level, int life, int score)

{

int n;

gotoxy(0,28);

printf("\nEnter the searching position\n");

gotoxy(0,29);

printf("1.Forward Search\n2.Backward Search\n3.Downward Search\n4.Upward Search\n5.Forward Upward(Diagonally) Search\n6.Backrward Upward(Diagonally) Search\n7.Forward Downward(Diagonally) Search\n8.Backward Downward(Diagonally) Search");

printf("\nsearching position is\n");

scanf("%d",&n);

if(n==1)

{

system("CLS");

printf("Forward Searchning\n");

Forward(l,o,level,life,score);

//score=score+10;

}

else if(n==2)

{

system("CLS");

printf("Backward Searchning\n");

Backward(l,o,level,life,score);

}

else if(n==3)

{

system("CLS");

printf("Downward Searchning\n");

Downward(l,o,level,life,score);

}

else if(n==4)

{

system("CLS");

printf("Upward Searchning\n");

Upward(l,o,level,life,score);

}

else if(n==5)

{

system("CLS");

printf("Forward Upward(Diagonally) Searchning\n");

ForwardUpward(l,o,level,life,score);

}

else if(n==6)

{

system("CLS");

printf("Backward Upward(Diagonally) Searchning\n");

BackwardUpward(l,o,level,life,score);

}

else if(n==7)

{

system("CLS");

printf("Forward Downward(Diagonally) Searchning\n");

ForwardDownward(l,o,level,life,score);

}

else if(n==8)

{

system("CLS");

printf("Backward Downward(Diagonally) Searchning\n");

BackwardDownward(l,o,level,life,score);

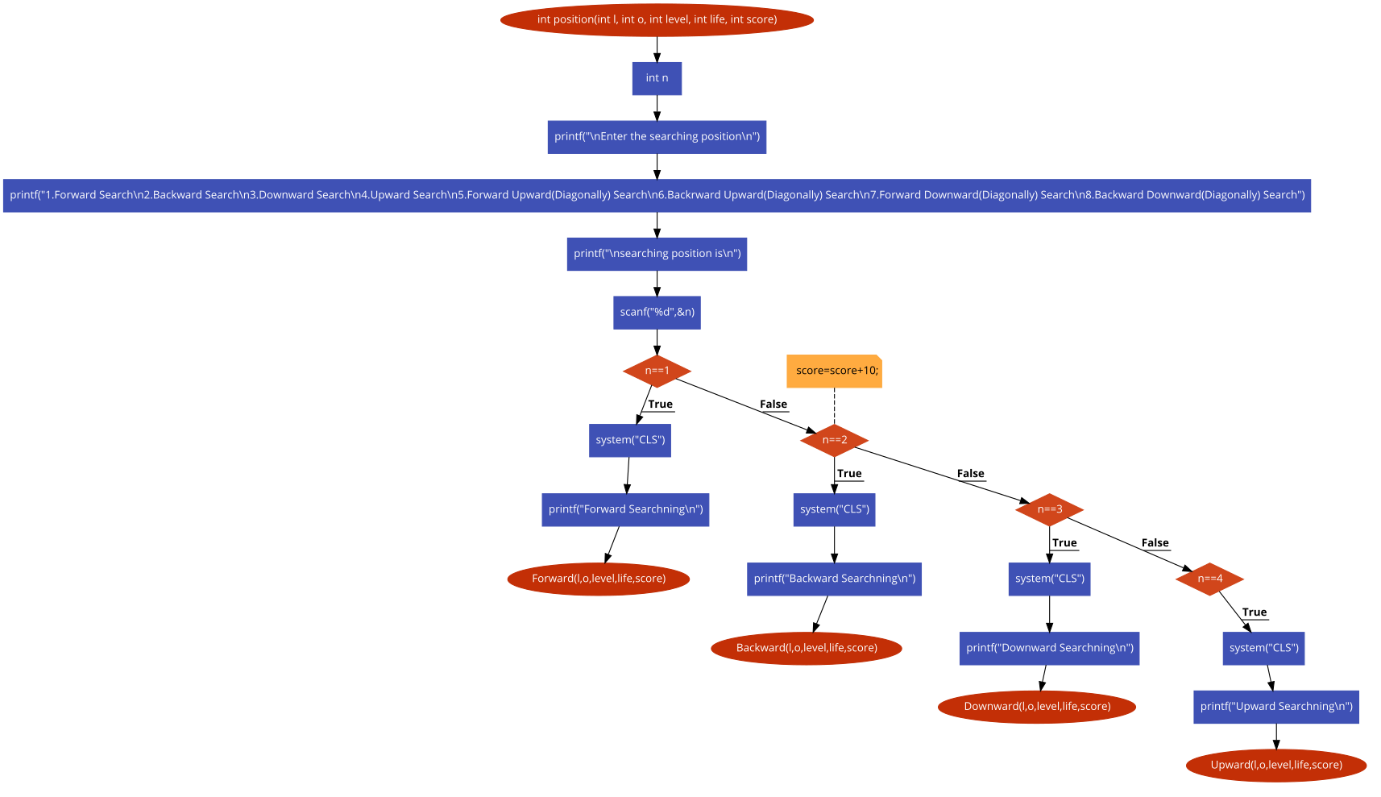
}

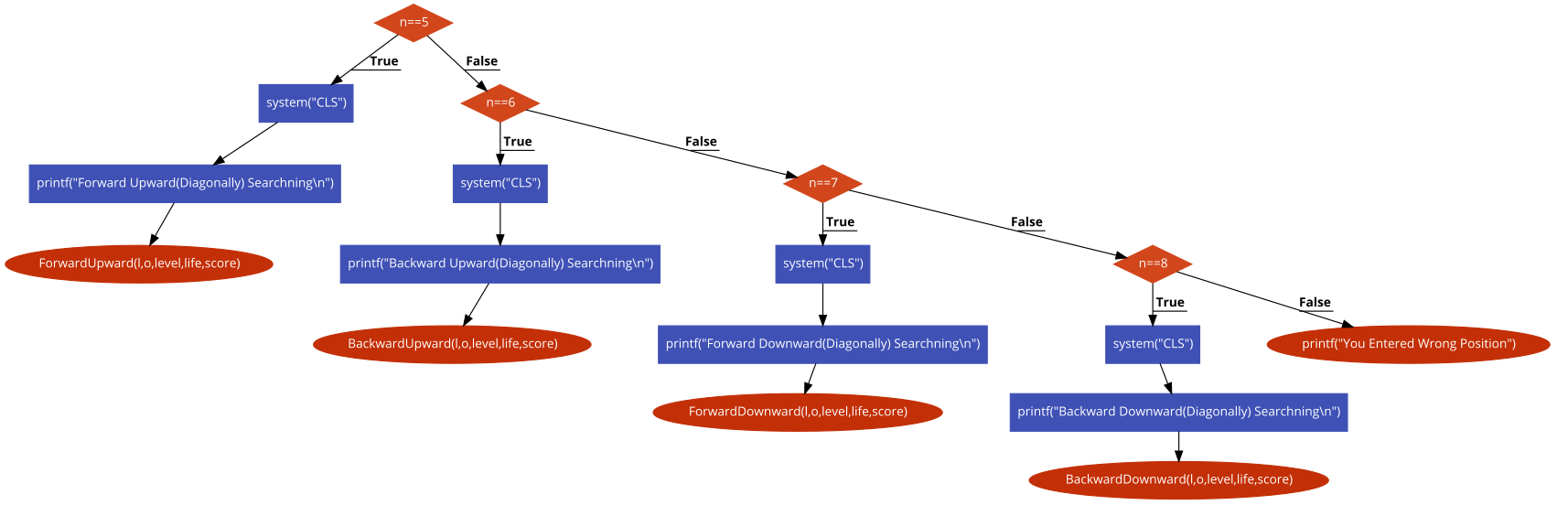
else

printf("You Entered Wrong Position");

}

**Flow chart: -**





**Function 11-18:-**

**int Forward(int l, int o, int level,int life, int score); //Function for Forward Searching of word within board**

**int Backward(int l, int o, int level,int life, int score); //Function for Backward Searching of word within board**

**int Downward(int l, int o, int level,int life, int score); //Function for Downward Searching of word within board**

**int Upward(int l, int o, int level,int life,int score); //Function for Upnward Searching of word within board**

**int ForwardUpward(int l, int o, int level,int life,int score); //Function for ForwardUpward Searching of word within board**

**int BackwardUpward(int l, int o,int level,int life, int score); //Function for BackwardUpward Searching of word within board**

**int ForwardDownward(int l, int o, int level,int life, int score); //Function for ForwardDownward Searching of word within board**

**int BackwardDownward(int l, int o, int level,int life, int score); //Function for BackwardDownward Searching of word within board**

After we entering the function of position we have the 8 options which we have to select at the run time so we select one of the position for searching also depends on the level according to given condition and the number which we select according to that we move forward in that function of selection statement and after doing that work we return to the position function back to select again until the specific given condition meets. The other three remaining functions are also call from these functions i.e. to store data in file to give the remaining life etc.

**Source code: -**

***//Forward Searching***

int Forward(int l, int o, int level,int life, int score)

{

struct ng e;

char ar[l][o],c,g;

int i,j,a,b,x,y,k=0,t=0,score1=0,n=0,q=0;

char ch[6]="camera",ch1[7]="comsats", ch2[3]="app",ch3[3]="boy",ch4[5]="apple";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the coloum\n");

scanf("%d",&b);

score1=0;

for(i=a; i<=a; i++)

{ //k=0;

for(j=x; j<=b; j++)

{

if(ar[i][j]==ch[k] || ch1[k] || ch2[k] || ch3[k] || ch4[k])

{

k++;

score1=score1 +1;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for Backward Searching of word within board***

int Backward(int l, int o, int level,int life,int score)

{

char ar[l][o],c,g;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[3]="car", ch1[4]="free";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the coloum\n");

scanf("%d",&b);

for(i=a; i<=a; i++)

{ k=0;

for(j=x; j>=b; j--)

{

if(ar[i][j]==ch[k] || ch1[k])

{

k++;

score1++;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for Downward Searching of word within board***

int Downward(int l, int o, int level,int life,int score)

{

char ar[l][o],c,g;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[4]="pack",ch1[3]="cyg";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the row\n");

scanf("%d",&b);

k=0;

for(i=a; i<=b; i++)

{

for(j=x; j<=x; j++)

{

if(ar[i][j]==ch[k] || ch1[k])

{

score1++;

k++;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for Upward Searching of word within board***

int Upward(int l, int o, int level,int life,int score)

{

char ar[l][o],c,g;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[3]="gas",ch1[6]="laptop";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the row\n");

scanf("%d",&b);

k=0;

for(i=a; i>=b; i--)

{

for(j=x; j<=x; j++)

{

if(ar[i][j]==ch[k] || ch1[k])

{

score1++;

k++;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for ForwardUpward Searching of word within board***

int ForwardUpward(int l, int o, int level,int life,int score)

{

char ar[l][o],c;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[3]="gas", ch1[6]="cooler";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the row\n");

scanf("%d",&b);

printf("And the Coloumn\n");

scanf("%d",&y);

k=0;

for(i=a; i>=b; i--)

{

for(j=x; j<=y; j++)

{

if(ar[i][j]==ch[k] || ch1[k])

{

score1++;

k++;

break;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for BackwardUpward Searching of word within board***

int BackwardUpward(int l, int o,int level,int life,int score)

{

char ar[l][o],c;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[3]="gas", ch1[5]="apple";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the row\n");

scanf("%d",&b);

printf("And the Coloumn\n");

scanf("%d",&y);

k=0;

for(i=a; i>=b; i--)

{

for(j=x; j>=y; j--)

{

if(ar[i][j]==ch[k] || ch1[k])

{

score1++;

k++;

break;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for ForwardDownward Searching of word within board***

int ForwardDownward(int l, int o, int level,int life,int score)

{

char ar[l][o],c;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[3]="app", ch1[3]="oil";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the row\n");

scanf("%d",&b);

printf("And the Coloumn\n");

scanf("%d",&y);

k=0;

for(i=a; i<=b; i++)

{

for(j=x; j<=y; j++)

{

if(ar[i][j]==ch[k] || ch1[k])

{

score1++;

k++;

break;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

}

}

***//Function for BackwardDownward Searching of word within board***

int BackwardDownward(int l, int o, int level,int life,int score)

{

char ar[l][o],c;

int i,j,a,b,x,y,k=0,score1=0,t=0,n=0,q=0;

char ch[4]="bulb", ch1[3]="tab";

read(level);

d:

score1=0;

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

boardstoring(l,o,level);

printf("Enter the starting position of your word\n");

printf("The word starts at the row\n");

scanf("%d",&a);

printf("And the coloum\n");

scanf("%d",&x);

printf("Enter the ending position of your given word\n");

printf("The word ends at the row\n");

scanf("%d",&b);

printf("And the Coloumn\n");

scanf("%d",&y);

k=0;

for(i=a; i<=b; i++)

{

for(j=x; j>=y; j--)

{

if(ar[i][j]==ch[k] || ch1[k])

{

score1++;

k++;

break;

}

}

}

if(score1>0 && t!=5)

{t=t+1;

score=score+10;

n=score2(score1,l,o,life,level,score,t);

if(n==1)

{

goto d;

}

}

if(score1==0);

{

q=life1(life,score1,level,t);

life=q;

goto d;

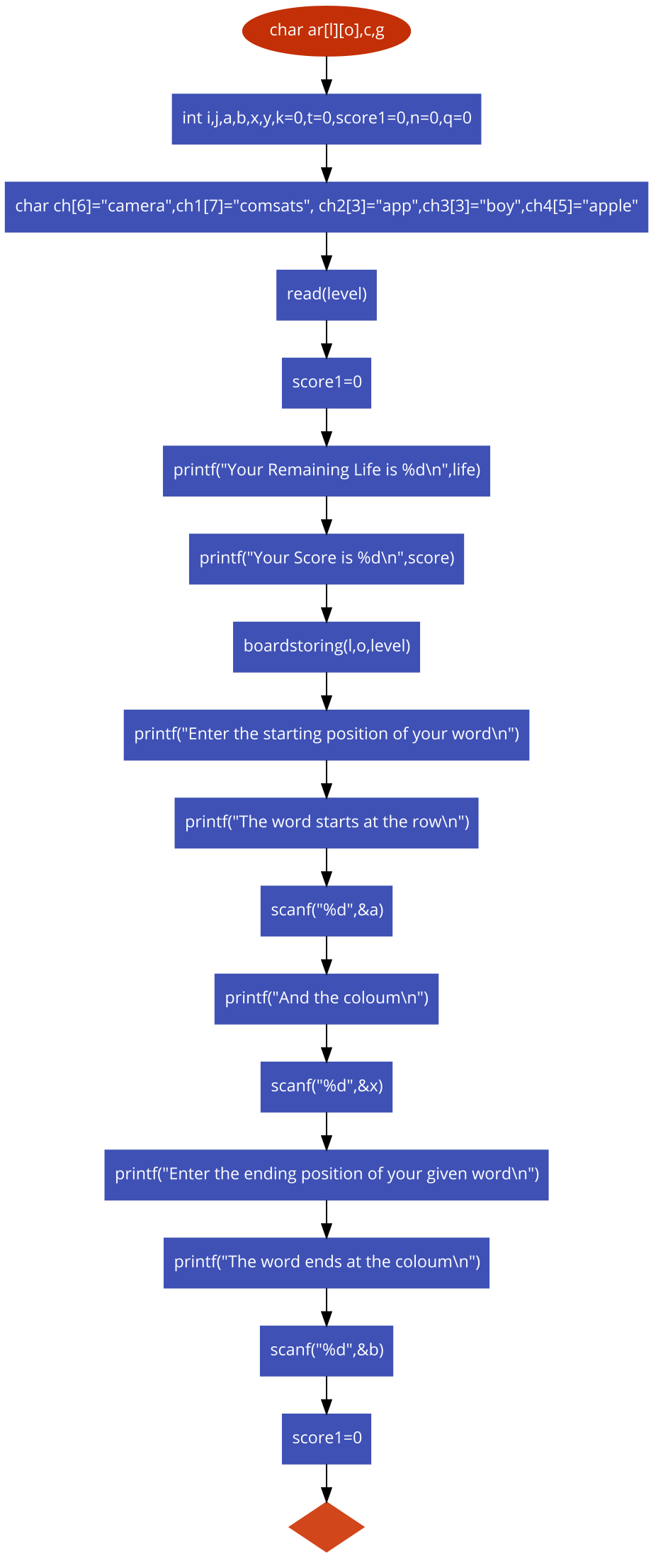
}

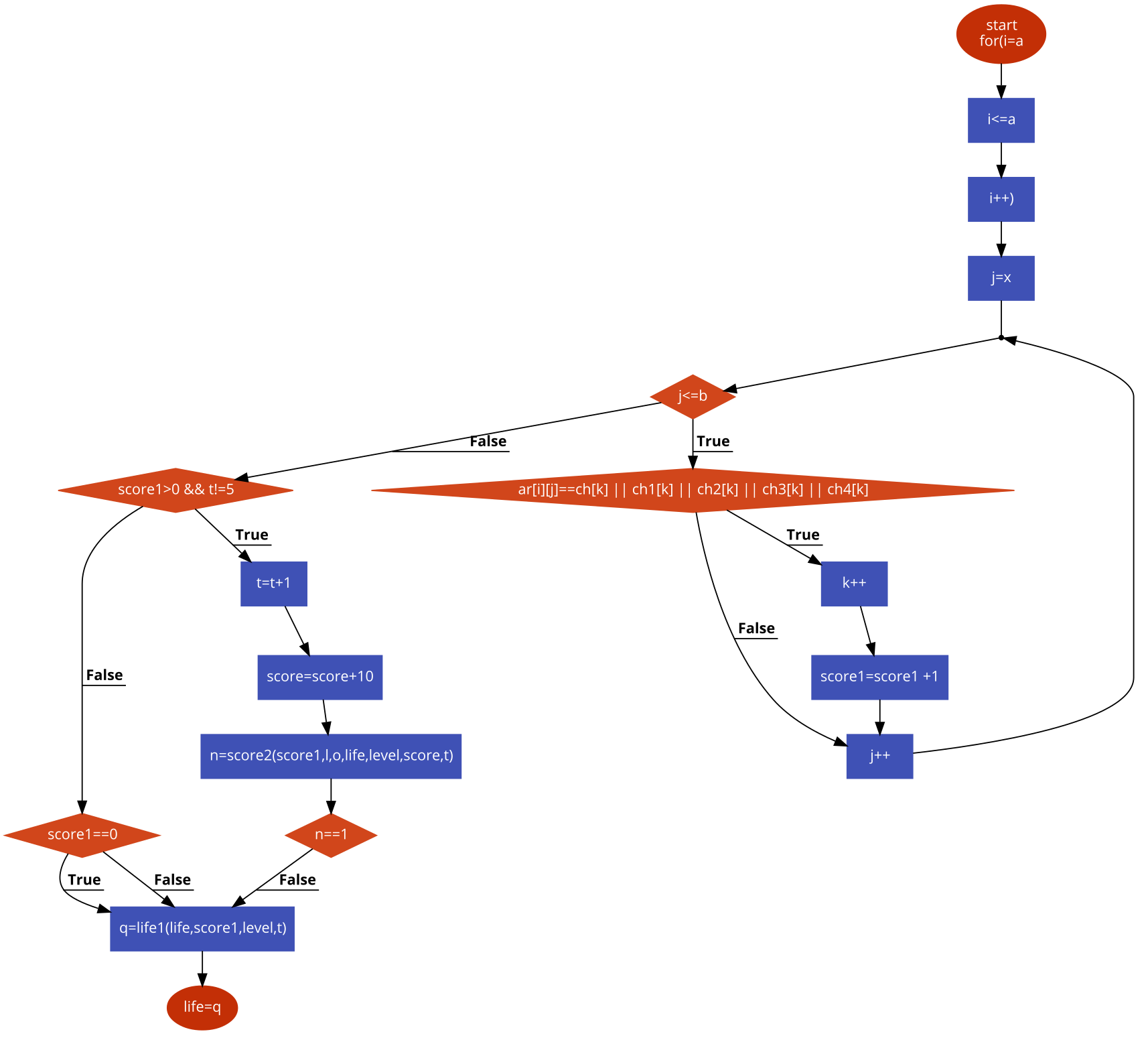
}

**Flow chart:-**

As source code is same for all of the above functions(11 to 18) so we make flow chart of any one

**Forward Search:-**





**Function 19**

**boardstoring(int l , int o, int level);**

This function takes the words from file and arrange them in a 2d array so that any word can be searched in 2d array using any position.

char boardstoring(int l , int o, int level)

{

char ar[l][o],c;

int i,j;

FILE \*fp;

if(level==1)

{fp=fopen("B11.txt","r");}

else if(level==2)

{fp=fopen("B12.txt","r");}

else if(level==3)

{fp=fopen("B13.txt","r");}

if(fp==NULL)

printf("File not Found");

for(i=0; i<l; i++)

{

for(j=0; j<o; j++)

{

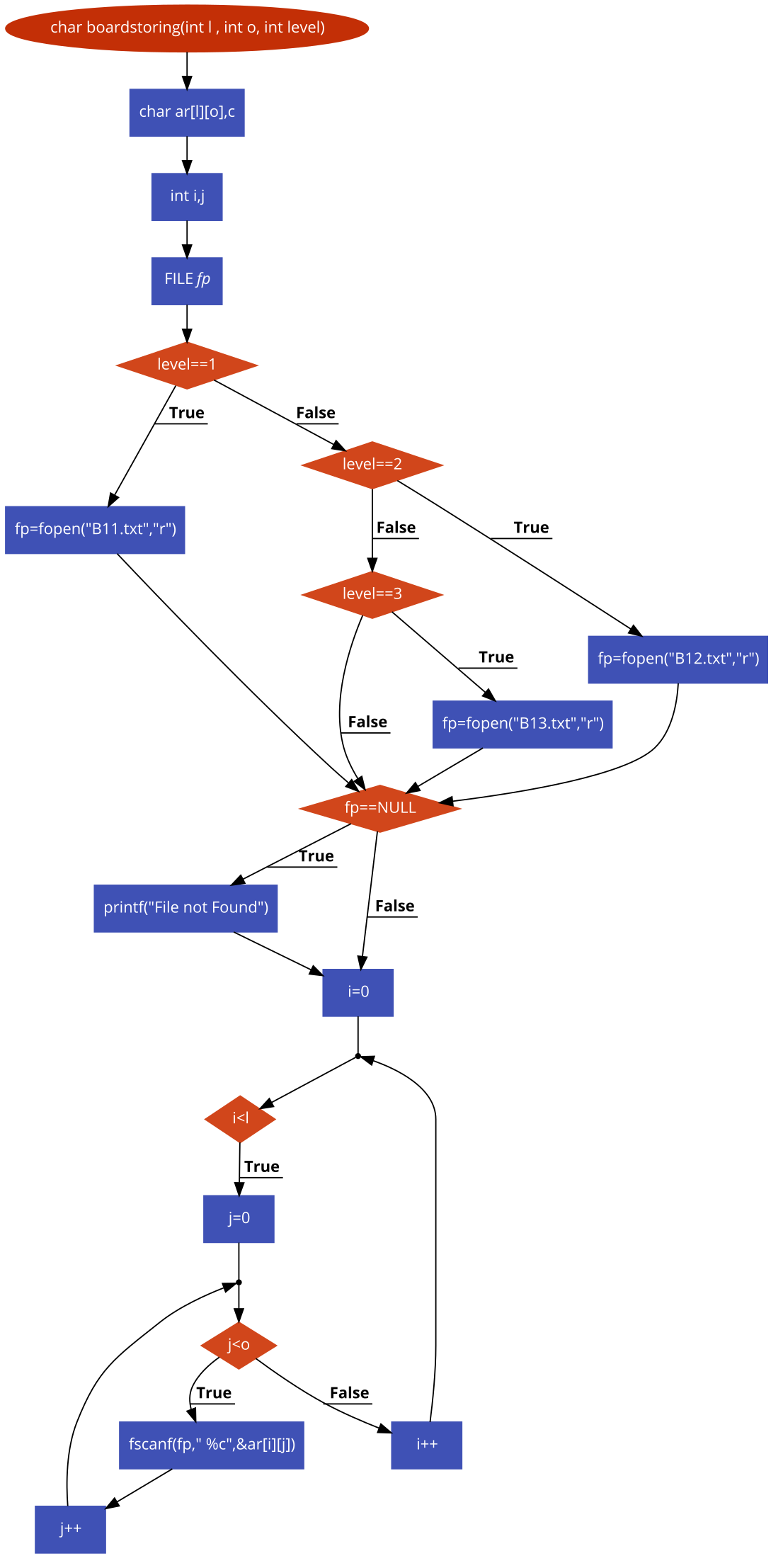
fscanf(fp," %c",&ar[i][j]); //The space in "..%c" consumes the newline character when using scanf() format specifiers such as %c %lf %d etc.

}

}

}

**Flowchart:**

****

**Function 20:**

**int score2(int score1,int l ,int o,int life,int level,int score, int t );**

This function is make for calculating the score of the player and tell us whether the user won the game or loss the game.

int score2(int score1,int l ,int o,int life,int level,int score, int t )

{

char c,g;

life=3;

if(score1>0)

{

printf("Congratulations word is found at given position\n");

if(t==5)

{

system("CLS");

gotoxy(60,15);

printf("YOU WON");

gotoxy(60,16);

printf("Press 1 to choose a Different SEARCHING POSTION or 2 to Choose a Different LEVEL or 3 to EXIT ");

c=getch();

if(c=='1')

{

position(l,o,level,life,score);

}

else if(c=='2')

{

main();

}

else if(c=='3')

{

exit(0);

}

}

sleep(1);

system("CLS");

read(level);

gotoxy(70,25);

printf("Your Remaining Life is %d\n",life);

gotoxy(70,27);

printf("Your Score is %d\n",score);

gotoxy(0,25);

printf("Do you want to change the Seraching Postion(Y/N)\n");

g=getch();

if(g=='Y' || g=='y')

position(l,o,level,life,score);

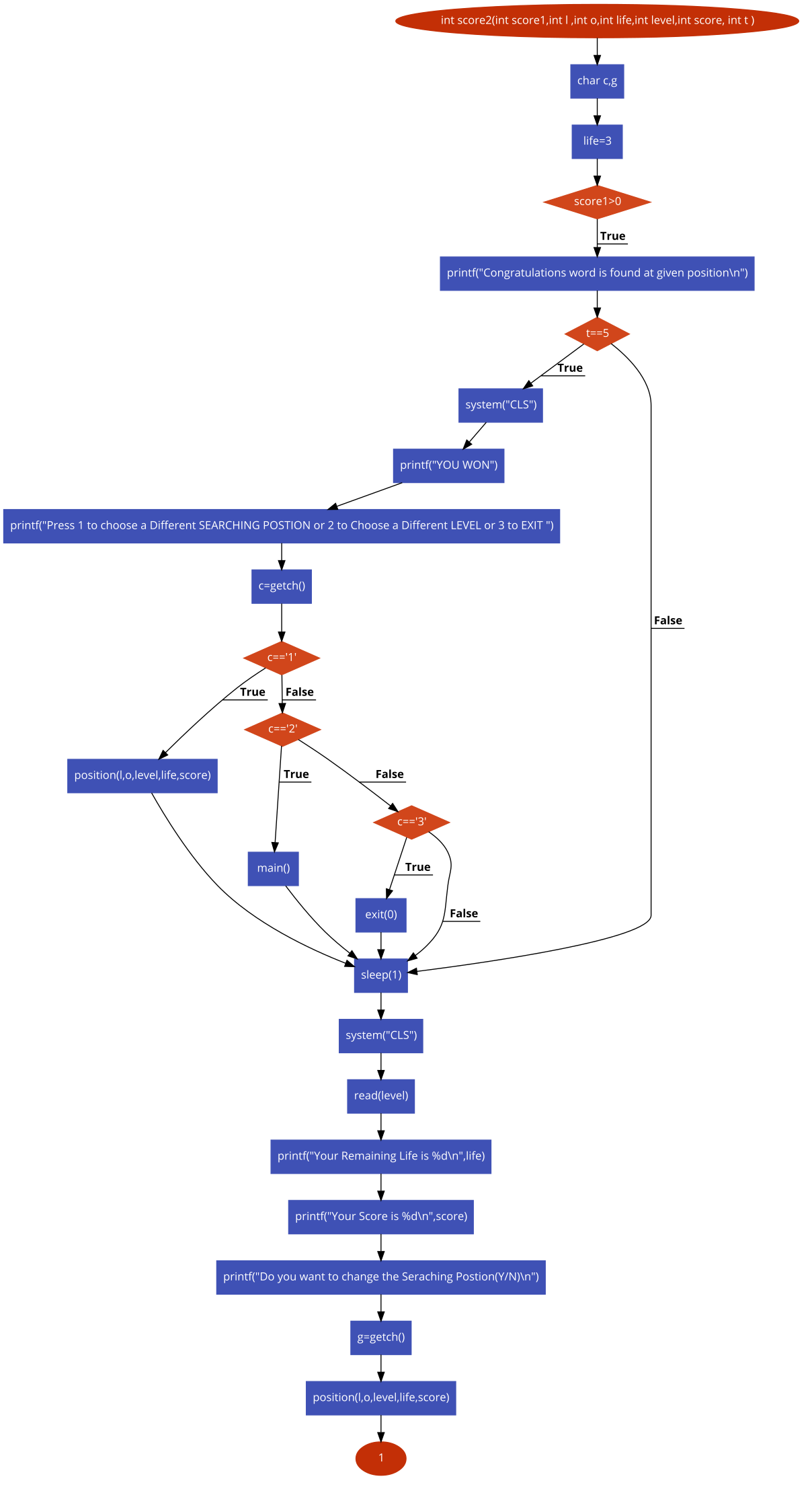
else if(g=='N' || g=='n')

return 1;

}

}

**Flowchart:**



**Function 21:**

**int life1(int life,int score1,int level,int t);**

This function calculates the life of player if player search a wrong word it a life is subtracted from it and if the player search consecutively 3 wrong words than he loses the game.

int life1(int life,int score1,int level,int t)

{

char c,g;

if(score1==0)

{

printf("Word not found");

sleep(1);

life=life-1;

if(life<=0)

{

system("CLS");

gotoxy(60,15);

printf("Life=0");

gotoxy(60,16);

printf("Game Over");

gotoxy(60,17);

printf("Press 1 to Play Again or 0 to Exit");

c=getch();

if(c=='1')

newgame(level);

else if(c=='0')

exit(0);

}

else if(t!=5)

{

system("CLS");

read(level);

return life;

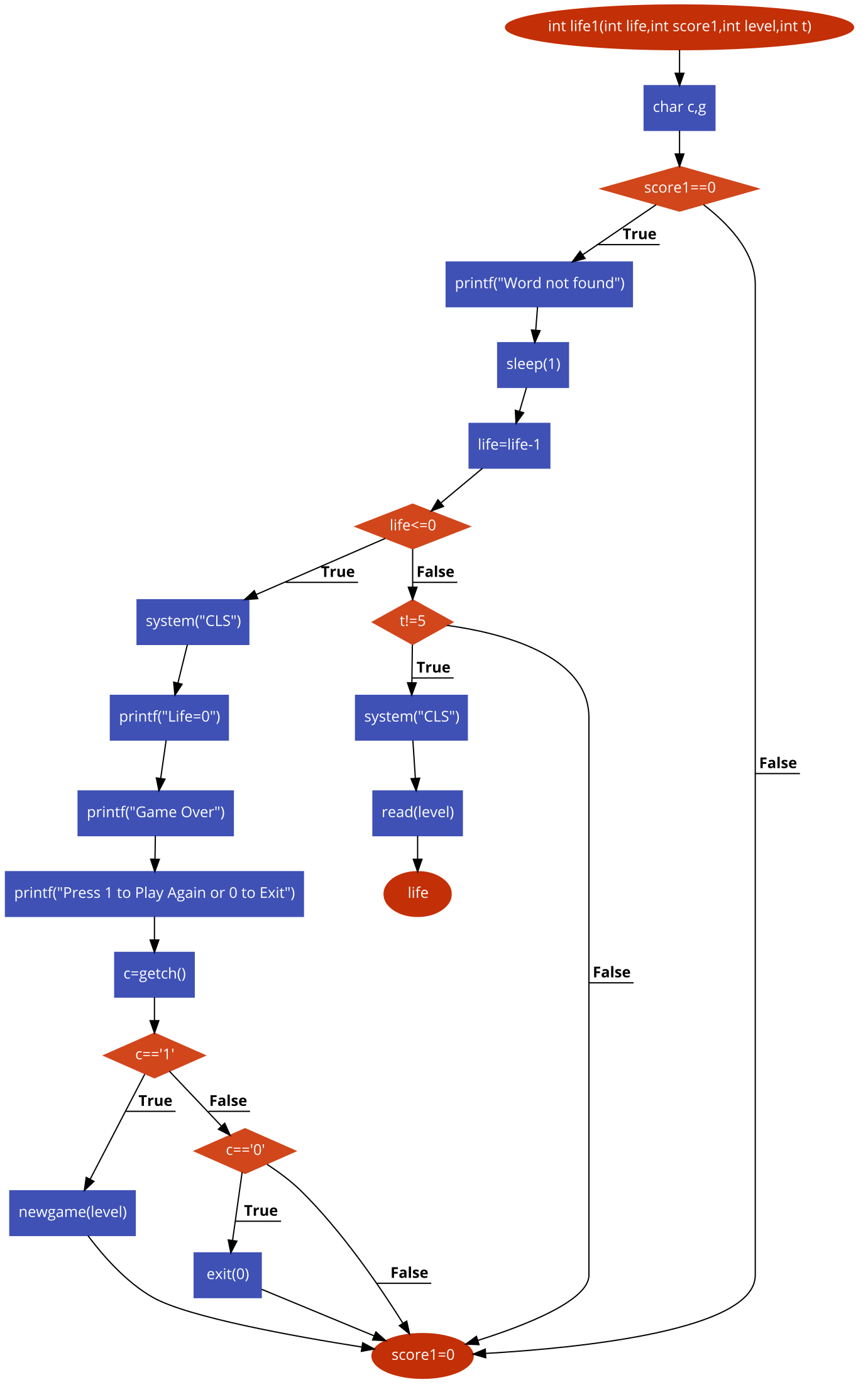
}

}

score1=0;

}

**Flowchart:**

****