// Name: Mehmet Fatih Çelik

// ID: 2385268

#include <stdio.h>

#include <stdlib.h>

int roll\_a\_dice(void);

int play\_computer(void);

int computer\_strategy\_decider(int, int, int, int, int, int, int, int, int);

int play\_user(void);

void scoresheet(int, int);

int main(){

int rounds, comp\_dice, your\_dice, i, first\_dice\_controller, score\_comp = 0, score\_your = 0, temp\_comp, temp\_your;

char choice;

srand(time(NULL));

printf("Welcome to the MidnightDice game.\n");

printf("Lets get started!\n\n");

printf("How many rounds would you like to play? ");

scanf("%d",&rounds);

do{ // if draw continue asking for input!

first\_dice\_controller = 0;

if(rounds> 0){

comp\_dice = roll\_a\_dice();

printf("I have rolled the dice and got %d\n",comp\_dice);

printf("Shall I roll the dice for you (Y/N)? ");

scanf(" %c",&choice);

if (choice == 'Y'){

your\_dice = roll\_a\_dice();

printf("I have rolled the dice for you and you got %d!\n",your\_dice);

}

if (your\_dice == comp\_dice){

printf("Draw! Let me roll again!\n");

first\_dice\_controller = 1; // for draw condition

}

}

}while(first\_dice\_controller);

for(i=1;i<=rounds;i++){

if (comp\_dice > your\_dice){

printf("\nRound %d -- My Turn:\n", i);

printf("=========================================================================\n");

temp\_comp = play\_computer();

score\_comp += temp\_comp;

printf("\nMy score: %d\n\n",temp\_comp);

printf("Round %d -- Your Turn:\n", i);

printf("=========================================================================\n");

temp\_your = play\_user();

score\_your += temp\_your;

printf("\nYour score: %d\n\n",temp\_your);

scoresheet(score\_comp, score\_your);

}

if (your\_dice > comp\_dice){

printf("Round %d -- Your Turn:\n", i);

printf("=========================================================================\n");

temp\_your = play\_user();

score\_your += temp\_your;

printf("\nYour score: %d\n\n",temp\_your);

printf("\nRound %d -- My Turn:\n", i);

printf("=========================================================================\n");

temp\_comp = play\_computer();

score\_comp += temp\_comp;

printf("\nMy score: %d\n\n",temp\_comp);

scoresheet(score\_comp, score\_your);

}

}

if (score\_comp > score\_your)

printf("I AM THE WINNER!");

else if (score\_your > score\_comp)

printf("YOU ARE THE WINNER!");

else

printf("DRAW!");

return 0;

}

int roll\_a\_dice(void){

return (1+ rand()%6);

}

int play\_computer(void){

int dice1, dice2, dice3, dice4, dice5, dice6, controller=0, controller\_1 = 0, controller\_4 = 0, score = 0;

int kept1, kept2, kept1\_2, kept2\_2, kept1\_3, kept2\_3; // first 2 for first 2 kept value, second 2 for second 2 kept value, third 2 for third 2 kept value

//dice1 = roll\_a\_dice();

//dice2 = roll\_a\_dice();

//dice3 = roll\_a\_dice();

//dice4 = roll\_a\_dice();

//dice5 = roll\_a\_dice();

//dice6 = roll\_a\_dice();

dice1 = 2;

dice2 = 4;

dice3 = 6;

dice4 = 5;

dice5 = 6;

dice6 = 2;

printf("I got -> [Dice 1]: %d [Dice 2]: %d [Dice 3]: %d [Dice 4]: %d [Dice 5]: %d [Dice 6]: %d\n",dice1,dice2,dice3,dice4,dice5,dice6);

kept1 = computer\_strategy\_decider(dice1, dice2, dice3, dice4, dice5, dice6, controller, controller\_1, controller\_4); // controller is 0, the function will return kept1

controller++;

kept2 = computer\_strategy\_decider(dice1, dice2, dice3, dice4, dice5, dice6, controller, controller\_1, controller\_4); // controller is 1, the function will return kept2

printf("Kept dice %d and %d\n",kept1, kept2);

if (kept1 == 1 || kept2 == 1){ //This 6 if blocks for "did we find 1 and 4" if 1 found, controller\_1 = 1, if 4 found, controller\_4 = 1

if (dice1 == 1)

controller\_1 = 1;

if (dice1 == 4)

controller\_4 = 1;

}

if (kept1 == 2 || kept2 == 2){

if (dice2 == 1)

controller\_1 = 1;

if (dice2 == 4)

controller\_4 = 1;

}

if (kept1 == 3 || kept2 == 3){

if (dice3 == 1)

controller\_1 = 1;

if (dice3 == 4)

controller\_4 = 1;

}

if (kept1 == 4 || kept2 == 4){

if (dice4 == 1)

controller\_1 = 1;

if (dice4 == 4)

controller\_4 = 1;

}

if (kept1 == 5 || kept2 == 5){

if (dice5 == 1)

controller\_1 = 1;

if (dice5 == 4)

controller\_4 = 1;

}

if (kept1 == 6 || kept2 == 6){

if (dice6 == 1)

controller\_1 = 1;

if (dice6 == 4)

controller\_4 = 1;

}

if (((controller\_1 == 1) && (controller\_4 == 0)) || ((controller\_1 == 0) && (controller\_4 == 1))){ // if we just find 1 or if we just find 4

if (kept2 == 1)

score += dice1;

if (kept2 == 2)

score += dice2;

if (kept2 == 3)

score += dice3;

if (kept2 == 4)

score += dice4;

if (kept2 == 5)

score += dice5;

if (kept2 == 6)

score += dice6;

}

else if ((controller\_1 == 0) && (controller\_4 == 0)){ // if we didnt find both 1 and 4

if ((kept1 == 1) || (kept2 == 1))

score += dice1;

if ((kept1 == 2) || (kept2 == 2))

score += dice2;

if ((kept1 == 3) || (kept2 == 3))

score += dice3;

if ((kept1 == 4) || (kept2 == 4))

score += dice4;

if ((kept1 == 5) || (kept2 == 5))

score += dice5;

if ((kept1 == 6) || (kept2 == 6))

score += dice6;

}

//dice1 = roll\_a\_dice();

//dice2 = roll\_a\_dice();

//dice3 = roll\_a\_dice();

//dice4 = roll\_a\_dice();

//dice5 = roll\_a\_dice();

//dice6 = roll\_a\_dice();

dice1 = 1;

dice1 = 2;

dice1 = 2;

dice1 = 5;

dice1 = 4;

dice1 = 3;

if (kept1 == 1 || kept2 == 1) // This 6 if statments for making 0 the previous kept dice value

dice1 = 0;

if (kept1 == 2 || kept2 == 2)

dice2 = 0;

if (kept1 == 3 || kept2 == 3)

dice3 = 0;

if (kept1 == 4 || kept2 == 4)

dice4 = 0;

if (kept1 == 5 || kept2 == 5)

dice5 = 0;

if (kept1 == 6 || kept2 == 6)

dice6 = 0;

printf("I got -> ");

if (dice1 != 0)

printf("[Dice 1]: %d ",dice1);

if (dice2 != 0)

printf("[Dice 2]: %d ",dice2);

if (dice3 != 0)

printf("[Dice 3]: %d ",dice3);

if (dice4 != 0)

printf("[Dice 4]: %d ",dice4);

if (dice5 != 0)

printf("[Dice 5]: %d ",dice5);

if (dice6 != 0)

printf("[Dice 6]: %d",dice6);

controller = 0;

kept1\_2 = computer\_strategy\_decider(dice1, dice2, dice3, dice4, dice5, dice6, controller, controller\_1, controller\_4); // controller is 0, the function will return kept1

controller++;

kept2\_2 = computer\_strategy\_decider(dice1, dice2, dice3, dice4, dice5, dice6, controller, controller\_1, controller\_4); // controller is 1, the function will return kept2

printf("\nKept dice %d and %d\n",kept1\_2, kept2\_2);

if (((controller\_1 == 1) && (controller\_4 == 0)) || ((controller\_1 == 0) && (controller\_4 == 1))){ // if we just find 1 or if we just find 4

if (kept2\_2 == 1)

score += dice1;

if (kept2\_2 == 2)

score += dice2;

if (kept2\_2 == 3)

score += dice3;

if (kept2\_2 == 4)

score += dice4;

if (kept2\_2 == 5)

score += dice5;

if (kept2\_2 == 6)

score += dice6;

}

else if ((controller\_1 == 1 && controller\_4 == 1) || (controller\_1 == 0 && controller\_4 == 0)) { // if we didnt find both 1 and 4

if ((kept1\_2 == 1) || (kept2\_2 == 1))

score += dice1;

if ((kept1\_2 == 2) || (kept2\_2 == 2))

score += dice2;

if ((kept1\_2 == 3) || (kept2\_2 == 3))

score += dice3;

if ((kept1\_2 == 4) || (kept2\_2 == 4))

score += dice4;

if ((kept1\_2 == 5) || (kept2\_2 == 5))

score += dice5;

if ((kept1\_2 == 6) || (kept2\_2 == 6))

score += dice6;

}

if (kept1\_2 == 1 || kept2\_2 == 1){ //This 6 if blocks for "did we find 1 and 4" if 1 found, controller\_1 = 1, if 4 found, controller\_4 = 1

if (dice1 == 1)

controller\_1 = 1;

if (dice1 == 4)

controller\_4 = 1;

}

if (kept1\_2 == 2 || kept2\_2 == 2){

if (dice2 == 1)

controller\_1 = 1;

if (dice2 == 4)

controller\_4 = 1;

}

if (kept1\_2 == 3 || kept2\_2 == 3){

if (dice3 == 1)

controller\_1 = 1;

if (dice3 == 4)

controller\_4 = 1;

}

if (kept1\_2 == 4 || kept2\_2 == 4){

if (dice4 == 1)

controller\_1 = 1;

if (dice4 == 4)

controller\_4 = 1;

}

if (kept1\_2 == 5 || kept2\_2 == 5){

if (dice5 == 1)

controller\_1 = 1;

if (dice5 == 4)

controller\_4 = 1;

}

if (kept1\_2 == 6 || kept2\_2 == 6){

if (dice6 == 1)

controller\_1 = 1;

if (dice6 == 4)

controller\_4 = 1;

}

dice1 = roll\_a\_dice();

dice2 = roll\_a\_dice();

dice3 = roll\_a\_dice();

dice4 = roll\_a\_dice();

dice5 = roll\_a\_dice();

dice6 = roll\_a\_dice();

if (kept1 == 1 || kept2 == 1 || kept1\_2 == 1 || kept2\_2 == 1)

dice1 = 0;

if (kept1 == 2 || kept2 == 2 || kept1\_2 == 2 || kept2\_2 == 2)

dice2 = 0;

if (kept1 == 3 || kept2 == 3 || kept1\_2 == 3 || kept2\_2 == 3)

dice3 = 0;

if (kept1 == 4 || kept2 == 4 || kept1\_2 == 4 || kept2\_2 == 4)

dice4 = 0;

if (kept1 == 5 || kept2 == 5 || kept1\_2 == 5 || kept2\_2 == 5)

dice5 = 0;

if (kept1 == 6 || kept2 == 6 || kept1\_2 == 6 || kept2\_2 == 6)

dice6 = 0;

controller = 0;

kept1\_3 = computer\_strategy\_decider(dice1, dice2, dice3, dice4, dice5, dice6, controller, controller\_1, controller\_4);

controller++;

kept2\_3 = computer\_strategy\_decider(dice1, dice2, dice3, dice4, dice5, dice6, controller, controller\_1, controller\_4);

if (((controller\_1 == 1) && (controller\_4 == 0)) || ((controller\_1 == 0) && (controller\_4 == 1))){ // if we just find 1 or if we just find 4

if (kept2\_3 == 1)

score += dice1;

if (kept2\_3 == 2)

score += dice2;

if (kept2\_3 == 3)

score += dice3;

if (kept2\_3 == 4)

score += dice4;

if (kept2\_3 == 5)

score += dice5;

if (kept2\_3 == 6)

score += dice6;

}

else if ((controller\_1 == 0 || controller\_1 == 1 ) && (controller\_4 == 0 || controller\_4 == 1)){ // if we didnt find both 1 and 4

if ((kept1\_3 == 1) || (kept2\_3 == 1))

score += dice1;

if ((kept1\_3 == 2) || (kept2\_3 == 2))

score += dice2;

if ((kept1\_3 == 3) || (kept2\_3 == 3))

score += dice3;

if ((kept1\_3 == 4) || (kept2\_3 == 4))

score += dice4;

if ((kept1\_3 == 5) || (kept2\_3 == 5))

score += dice5;

if ((kept1\_3 == 6) || (kept2\_3 == 6))

score += dice6;

}

if (kept1\_3 == 1 || kept2\_3 == 1){ //This 6 if blocks for "did we find 1 and 4" if 1 found, controller\_1 = 1, if 4 found, controller\_4 = 1

if (dice1 == 1)

controller\_1 = 1;

if (dice1 == 4)

controller\_4 = 1;

}

if (kept1\_3 == 2 || kept2\_3 == 2){

if (dice2 == 1)

controller\_1 = 1;

if (dice2 == 4)

controller\_4 = 1;

}

if (kept1\_3 == 3 || kept2\_3 == 3){

if (dice3 == 1)

controller\_1 = 1;

if (dice3 == 4)

controller\_4 = 1;

}

if (kept1\_3 == 4 || kept2\_3 == 4){

if (dice4 == 1)

controller\_1 = 1;

if (dice4 == 4)

controller\_4 = 1;

}

if (kept1\_3 == 5 || kept2\_3 == 5){

if (dice5 == 1)

controller\_1 = 1;

if (dice5 == 4)

controller\_4 = 1;

}

if (kept1\_3 == 6 || kept2\_3 == 6){

if (dice6 == 1)

controller\_1 = 1;

if (dice6 == 4)

controller\_4 = 1;

}

printf("I got -> ");

if (dice1 != 0)

printf("[Dice 1]: %d ",dice1);

if (dice2 != 0)

printf("[Dice 2]: %d ",dice2);

if (dice3 != 0)

printf("[Dice 3]: %d ",dice3);

if (dice4 != 0)

printf("[Dice 4]: %d ",dice4);

if (dice5 != 0)

printf("[Dice 5]: %d ",dice5);

if (dice6 != 0)

printf("[Dice 6]: %d",dice6);

if(controller\_1 == 0 || controller\_4 == 0) // if 4 or 1 is not found, score is 0!

score = 0;

return score;

}

int computer\_strategy\_decider(int dice1, int dice2, int dice3, int dice4, int dice5, int dice6, int controller, int controller\_1, int controller\_4){

int kept1=9, kept2, total\_kept=0, biggest = 0, biggest2=0; // kept1 equals an arbitrary value because later, I will compare with values to learn if we get kept1.

if ((dice1 == 1 || dice1 == 4) && (dice1 != 0) && (!controller\_1 || !controller\_4)){ // if dice is 1 or 4 and dice hadnt been kept before!

if(controller\_1 == 0 && dice1 == 1){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 1;

else

kept1 = 1;

controller\_1 = 1;

total\_kept++;

}

else if(controller\_4 == 0 && dice1 == 4){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 1;

else

kept1 = 1;

controller\_4 = 1;

total\_kept++;

}

}

if ((dice2 == 1 || dice2 == 4) && (dice2 != 0) && (!controller\_1 || !controller\_4)){ // if dice is 1 or 4 and dice hadnt been kept before!

if(controller\_1 == 0 && dice2 == 1){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 2;

else

kept1 = 2;

controller\_1 = 1;

total\_kept++;

}

else if(controller\_4 == 0 && dice2 == 4){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 2;

else

kept1 = 2;

controller\_4 = 1;

total\_kept++;

}

}

if ((dice3 == 1 || dice3 == 4) && (dice3 != 0) && (!controller\_1 || !controller\_4)){ // if dice is 1 or 4 and dice hadnt been kept before!

if(controller\_1 == 0 && dice3 == 1){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 3;

else

kept1 = 3;

controller\_1 = 1;

total\_kept++;

}

else if(controller\_4 == 0 && dice3 == 4){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 3;

else

kept1 = 3;

controller\_4 = 1;

total\_kept++;

}

}

if ((dice4 == 1 || dice4 == 4) && (dice4 != 0) && (!controller\_1 || !controller\_4)){ // if dice is 1 or 4 and dice hadnt been kept before!

if(controller\_1 == 0 && dice4 == 1){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 4;

else

kept1 = 4;

controller\_1 = 1;

total\_kept++;

}

else if(controller\_4 == 0 && dice4 == 4){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 4;

else

kept1 = 4;

controller\_4 = 1;

total\_kept++;

}

}

if ((dice5 == 1 || dice5 == 4) && (dice5 != 0) && (!controller\_1 || !controller\_4)){ // if dice is 1 or 4 and dice hadnt been kept before!

if(controller\_1 == 0 && dice5 == 1){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 5;

else

kept1 = 5;

controller\_1 = 1;

total\_kept++;

}

else if(controller\_4 == 0 && dice5 == 4){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 5;

else

kept1 = 5;

controller\_4 = 1;

total\_kept++;

}

}

if ((dice6 == 1 || dice6 == 4) && (dice6 != 0) && (!controller\_1 || !controller\_4)){ // if dice is 1 or 4 and dice hadnt been kept before!

if(controller\_1 == 0 && dice6 == 1){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 6;

else

kept1 = 6;

controller\_1 = 1;

total\_kept++;

}

else if(controller\_4 == 0 && dice6 == 4){

if (kept1 == 1 || kept1 == 2 || kept1 == 3 || kept1 == 4 || kept1 == 5 || kept1 == 6)

kept2 = 6;

else

kept1 = 6;

controller\_4 = 1;

total\_kept++;

}

}

if(total\_kept == 1){ // if we found any of 1 or 4

if(dice1 != 0)

kept2 = 1;

if (((biggest < dice1) && (kept1!=1)) && (dice1 != 0)){

kept2 = 1;

biggest = dice1;

}

if (((biggest < dice2) && (kept1!=2)) && (dice2 != 0)){

kept2 = 2;

biggest = dice2;

}

if (((biggest < dice3) && (kept1!=3)) && (dice3 != 0)){

kept2 = 3;

biggest = dice3;

}

if (((biggest < dice4) && (kept1!=4)) && (dice4 != 0)){

kept2 = 4;

biggest = dice4;

}

if (((biggest < dice5) && (kept1!=5)) && (dice5 != 0)){

kept2 = 5;

biggest = dice5;

}

if (((biggest < dice6) && (kept1!=6)) && (dice6 != 0)){

kept2 = 6;

biggest = dice6;

}

total\_kept++;

}

else if (total\_kept == 0){ // if we didnt find any 1 or 4

if ((dice1 > biggest) && (dice1 != 0)){ //Simple algorithm that I wrote for finding maxs of two values without using arrays, pointers. It lasts to line 701.

biggest2 = biggest;

biggest = dice1;

kept1 = 1;

kept2 = 0;

}

else if ((dice1 > biggest2) && (dice1 != 0)){

biggest2 = dice1;

kept2 = 1;

}

if ((dice2 > biggest) && (dice2 != 0)){

biggest2 = biggest;

biggest = dice2;

kept1 = 2;

kept2 = 1;

}

else if ((dice2 > biggest2) && (dice2 != 0)){

biggest2 = dice2;

kept2 = 2;

}

if ((dice3 > biggest) && (dice3 != 0)){

biggest2 = biggest;

biggest = dice3;

kept2 = kept1;

kept1 = 3;

}

else if ((dice3 > biggest2) && (dice3 != 0)){

biggest2 = dice3;

kept2 = 3;

}

if ((dice4 > biggest) && (dice4 != 0)){

biggest2 = biggest;

biggest = dice4;

kept2 = kept1;

kept1 = 4;

}

else if ((dice4 > biggest2) && (dice4 != 0)){

biggest2 = dice4;

kept2 = 4;

}

if ((dice5 > biggest) && (dice5 != 0)){

biggest2 = biggest;

biggest = dice5;

kept2 = kept1;

kept1 = 5;

}

else if ((dice5 > biggest2) && (dice5 != 0)){

biggest2 = dice5;

kept2 = 5;

}

if ((dice6 > biggest) && (dice6 != 0)){

biggest2 = biggest;

biggest = dice6;

kept2 = kept1;

kept1 = 6;

}

else if ((dice6 > biggest2) && (dice6 != 0)){

biggest2 = dice6;

kept2 = 6;

}

}

if(controller == 0)

return kept1;

else if (controller == 1)

return kept2;

}

int play\_user(void){

int dice1, dice2, dice3, dice4, dice5, dice6, controller\_1 = 0, controller\_4 = 0;

int keepIndex\_1, keepIndex\_2, score = 0, keepIndex\_1\_2, keepIndex\_2\_2, keepIndex\_1\_3 = 9, keepIndex\_2\_3;

char choice;

printf("Are you ready to play!\n");

fflush(stdin);

printf("Shall I roll them for you (Y/N)? ");

scanf("%c",&choice);

if(choice == 'N')

return 0;

else if (choice == 'Y'){

dice1 = roll\_a\_dice();

dice2 = roll\_a\_dice();

dice3 = roll\_a\_dice();

dice4 = roll\_a\_dice();

dice5 = roll\_a\_dice();

dice6 = roll\_a\_dice();

printf("You got -> [Dice 1]: %d [Dice 2]: %d [Dice 3]: %d [Dice 4]: %d [Dice 5]: %d [Dice 6]: %d\n", dice1, dice2, dice3, dice4, dice5, dice6);

printf("Which ones you want to keep? ");

scanf("%d %d",&keepIndex\_1,&keepIndex\_2);

if (keepIndex\_1 == 1 || keepIndex\_2 == 1){ // controlling for if we kept 1 or 4, if we did change the flag! It lasts to line 765

if (dice1 == 1)

controller\_1 = 1;

else if (dice1 == 4)

controller\_4 = 1;

}

if (keepIndex\_1 == 2 || keepIndex\_2 == 2){

if (dice2 == 1)

controller\_1 = 1;

else if (dice2 == 4)

controller\_4 = 1;

}

if (keepIndex\_1 == 3 || keepIndex\_2 == 3){

if (dice3 == 1)

controller\_1 = 1;

else if (dice3 == 4)

controller\_4 = 1;

}

if (keepIndex\_1 == 4 || keepIndex\_2 == 4){

if (dice4 == 1)

controller\_1 = 1;

else if (dice4 == 4)

controller\_4 = 1;

}

if (keepIndex\_1 == 5 || keepIndex\_2 == 5){

if (dice5 == 1)

controller\_1 = 1;

else if (dice5 == 4)

controller\_4 = 1;

}

if (keepIndex\_1 == 6 || keepIndex\_2 == 6){

if (dice6 == 1)

controller\_1 = 1;

else if (dice6 == 4)

controller\_4 = 1;

}

if (((controller\_1 == 1) && (controller\_4 == 0)) || ((controller\_1 == 0) && (controller\_4 == 1))){ // if user just selects one dice which value's is 1 or 4

if((keepIndex\_1 == 1 && dice1 == 1) || (keepIndex\_1 == 1 && dice1 == 4)){ // score increasing appropriately. It last to line 931.

if (keepIndex\_2 == 2)

score += dice2;

if (keepIndex\_2 == 3)

score += dice3;

if (keepIndex\_2 == 4)

score += dice4;

if (keepIndex\_2 == 5)

score += dice5;

if (keepIndex\_2 == 6)

score += dice6;

}

else if((keepIndex\_2 == 1 && dice1 == 1) || (keepIndex\_2 == 1 && dice1 == 4)){

if (keepIndex\_1 == 2)

score += dice2;

if (keepIndex\_1 == 3)

score += dice3;

if (keepIndex\_1 == 4)

score += dice4;

if (keepIndex\_1 == 5)

score += dice5;

if (keepIndex\_1 == 6)

score += dice6;

}

if((keepIndex\_1 == 2 && dice2 == 1) || (keepIndex\_1 == 2 && dice2 == 4)){

if (keepIndex\_2 == 1)

score += dice1;

if (keepIndex\_2 == 3)

score += dice3;

if (keepIndex\_2 == 4)

score += dice4;

if (keepIndex\_2 == 5)

score += dice5;

if (keepIndex\_2 == 6)

score += dice6;

}

else if((keepIndex\_2 == 2 && dice2 == 1) || (keepIndex\_2 == 2 && dice2 == 4)){

if (keepIndex\_1 == 1)

score += dice1;

if (keepIndex\_1 == 3)

score += dice3;

if (keepIndex\_1 == 4)

score += dice4;

if (keepIndex\_1 == 5)

score += dice5;

if (keepIndex\_1 == 6)

score += dice6;

}

if((keepIndex\_1 == 3 && dice3 == 1) || (keepIndex\_1 == 3 && dice3 == 4)){

if (keepIndex\_2 == 1)

score += dice1;

if (keepIndex\_2 == 2)

score += dice2;

if (keepIndex\_2 == 4)

score += dice4;

if (keepIndex\_2 == 5)

score += dice5;

if (keepIndex\_2 == 6)

score += dice6;

}

else if((keepIndex\_2 == 3 && dice3 == 1) || (keepIndex\_2 == 3 && dice3 == 4)){

if (keepIndex\_1 == 1)

score += dice1;

if (keepIndex\_1 == 2)

score += dice2;

if (keepIndex\_1 == 4)

score += dice4;

if (keepIndex\_1 == 5)

score += dice5;

if (keepIndex\_1 == 6)

score += dice6;

}

if((keepIndex\_1 == 4 && dice4 == 1) || (keepIndex\_1 == 4 && dice4 == 4)){

if (keepIndex\_2 == 1)

score += dice1;

if (keepIndex\_2 == 2)

score += dice2;

if (keepIndex\_2 == 3)

score += dice3;

if (keepIndex\_2 == 5)

score += dice5;

if (keepIndex\_2 == 6)

score += dice6;

}

else if((keepIndex\_2 == 4 && dice4 == 1) || (keepIndex\_2 == 4 && dice4 == 4)){

if (keepIndex\_1 == 1)

score += dice1;

if (keepIndex\_1 == 2)

score += dice2;

if (keepIndex\_1 == 3)

score += dice3;

if (keepIndex\_1 == 5)

score += dice5;

if (keepIndex\_1 == 6)

score += dice6;

}

if((keepIndex\_1 == 5 && dice5 == 1) || (keepIndex\_1 == 5 && dice5 == 4)){

if (keepIndex\_2 == 1)

score += dice1;

if (keepIndex\_2 == 2)

score += dice2;

if (keepIndex\_2 == 3)

score += dice3;

if (keepIndex\_2 == 4)

score += dice4;

if (keepIndex\_2 == 6)

score += dice6;

}

else if((keepIndex\_2 == 5 && dice5 == 1) || (keepIndex\_2 == 5 && dice5 == 4)){

if (keepIndex\_1 == 1)

score += dice1;

if (keepIndex\_1 == 2)

score += dice2;

if (keepIndex\_1 == 3)

score += dice3;

if (keepIndex\_1 == 4)

score += dice4;

if (keepIndex\_1 == 6)

score += dice6;

}

if((keepIndex\_1 == 6 && dice6 == 1) || (keepIndex\_1 == 6 && dice6 == 4)){

if (keepIndex\_2 == 1)

score += dice1;

if (keepIndex\_2 == 2)

score += dice2;

if (keepIndex\_2 == 3)

score += dice3;

if (keepIndex\_2 == 4)

score += dice4;

if (keepIndex\_2 == 5)

score += dice5;

}

else if((keepIndex\_2 == 6 && dice6 == 1) || (keepIndex\_2 == 6 && dice6 == 4)){

if (keepIndex\_1 == 1)

score += dice1;

if (keepIndex\_1 == 2)

score += dice2;

if (keepIndex\_1 == 3)

score += dice3;

if (keepIndex\_1 == 4)

score += dice4;

if (keepIndex\_1 == 5)

score += dice5;

}

}

if(controller\_1 == 0 && controller\_4 == 0){ // the condition that user didnt choose 1 and 4

if(keepIndex\_1 == 1 || keepIndex\_2 == 1)

score += dice1;

if(keepIndex\_1 == 2 || keepIndex\_2 == 2)

score += dice2;

if(keepIndex\_1 == 3 || keepIndex\_2 == 3)

score += dice3;

if(keepIndex\_1 == 4 || keepIndex\_2 == 4)

score += dice4;

if(keepIndex\_1 == 5 || keepIndex\_2 == 5)

score += dice5;

if(keepIndex\_1 == 6 || keepIndex\_2 == 6)

score += dice6;

}

fflush(stdin);

printf("Shall I roll the remaining for you (Y/N)? ");

scanf("%c",&choice);

if (choice == 'N'){

if (controller\_1 == 1 && controller\_4 == 1) // if we both find 1 and 4 already, return score, if not return 0

return score;

else

return 0;

}

else if (choice == 'Y'){

dice1 = roll\_a\_dice();

dice2 = roll\_a\_dice();

dice3 = roll\_a\_dice();

dice4 = roll\_a\_dice();

dice5 = roll\_a\_dice();

dice6 = roll\_a\_dice();

if (keepIndex\_1 == 1 || keepIndex\_2 == 1) // for making 0 for the previous kept dice values

dice1 = 0;

if (keepIndex\_1 == 2 || keepIndex\_2 == 2)

dice2 = 0;

if (keepIndex\_1 == 3 || keepIndex\_2 == 3)

dice3 = 0;

if (keepIndex\_1 == 4 || keepIndex\_2 == 4)

dice4 = 0;

if (keepIndex\_1 == 5 || keepIndex\_2 == 5)

dice5 = 0;

if (keepIndex\_1 == 6 || keepIndex\_2 == 6)

dice6 = 0;

printf("You got -> ");

if (dice1 != 0)

printf("[Dice 1]: %d ",dice1);

if (dice2 != 0)

printf("[Dice 2]: %d ",dice2);

if (dice3 != 0)

printf("[Dice 3]: %d ",dice3);

if (dice4 != 0)

printf("[Dice 4]: %d ",dice4);

if (dice5 != 0)

printf("[Dice 5]: %d ",dice5);

if (dice6 != 0)

printf("[Dice 6]: %d",dice6);

printf("\nWhich ones you want to keep? ");

scanf("%d %d",&keepIndex\_1\_2,&keepIndex\_2\_2);

if (((controller\_1 == 1) && (controller\_4 == 0)) || ((controller\_1 == 0) && (controller\_4 == 1))){ // if user just selects one dice which value's is 1 or 4

if((keepIndex\_1\_2 == 1 && dice1 == 1) || (keepIndex\_1\_2 == 1 && dice1 == 4)){//I used the same algorithm that I used above.

if (keepIndex\_2\_2 == 2)

score += dice2;

if (keepIndex\_2\_2 == 3)

score += dice3;

if (keepIndex\_2\_2 == 4)

score += dice4;

if (keepIndex\_2\_2 == 5)

score += dice5;

if (keepIndex\_2\_2 == 6)

score += dice6;

}

else if((keepIndex\_2\_2 == 1 && dice1 == 1) || (keepIndex\_2\_2 == 1 && dice1 == 4)){

if (keepIndex\_1\_2 == 2)

score += dice2;

if (keepIndex\_1\_2 == 3)

score += dice3;

if (keepIndex\_1\_2 == 4)

score += dice4;

if (keepIndex\_1\_2 == 5)

score += dice5;

if (keepIndex\_1\_2 == 6)

score += dice6;

}

if((keepIndex\_1\_2 == 2 && dice2 == 1) || (keepIndex\_1\_2 == 2 && dice2 == 4)){

if (keepIndex\_2\_2 == 1)

score += dice1;

if (keepIndex\_2\_2 == 3)

score += dice3;

if (keepIndex\_2\_2 == 4)

score += dice4;

if (keepIndex\_2\_2 == 5)

score += dice5;

if (keepIndex\_2\_2 == 6)

score += dice6;

}

else if((keepIndex\_2\_2 == 2 && dice2 == 1) || (keepIndex\_2\_2 == 2 && dice2 == 4)){

if (keepIndex\_1\_2 == 1)

score += dice1;

if (keepIndex\_1\_2 == 3)

score += dice3;

if (keepIndex\_1\_2 == 4)

score += dice4;

if (keepIndex\_1\_2 == 5)

score += dice5;

if (keepIndex\_1\_2 == 6)

score += dice6;

}

if((keepIndex\_1\_2 == 3 && dice3 == 1) || (keepIndex\_1\_2 == 3 && dice3 == 4)){

if (keepIndex\_2\_2 == 1)

score += dice1;

if (keepIndex\_2\_2 == 2)

score += dice2;

if (keepIndex\_2\_2 == 4)

score += dice4;

if (keepIndex\_2\_2 == 5)

score += dice5;

if (keepIndex\_2\_2 == 6)

score += dice6;

}

else if((keepIndex\_2\_2 == 3 && dice3 == 1) || (keepIndex\_2\_2 == 3 && dice3 == 4)){

if (keepIndex\_1\_2 == 1)

score += dice1;

if (keepIndex\_1\_2 == 2)

score += dice2;

if (keepIndex\_1\_2 == 4)

score += dice4;

if (keepIndex\_1\_2 == 5)

score += dice5;

if (keepIndex\_1\_2 == 6)

score += dice6;

}

if((keepIndex\_1\_2 == 4 && dice4 == 1) || (keepIndex\_1\_2 == 4 && dice4 == 4)){

if (keepIndex\_2\_2 == 1)

score += dice1;

if (keepIndex\_2\_2 == 2)

score += dice2;

if (keepIndex\_2\_2 == 3)

score += dice3;

if (keepIndex\_2\_2 == 5)

score += dice5;

if (keepIndex\_2\_2 == 6)

score += dice6;

}

else if((keepIndex\_2\_2 == 4 && dice4 == 1) || (keepIndex\_2\_2 == 4 && dice4 == 4)){

if (keepIndex\_1\_2 == 1)

score += dice1;

if (keepIndex\_1\_2 == 2)

score += dice2;

if (keepIndex\_1\_2 == 3)

score += dice3;

if (keepIndex\_1\_2 == 5)

score += dice5;

if (keepIndex\_1\_2 == 6)

score += dice6;

}

if((keepIndex\_1\_2 == 5 && dice5 == 1) || (keepIndex\_1\_2 == 5 && dice5 == 4)){

if (keepIndex\_2\_2 == 1)

score += dice1;

if (keepIndex\_2\_2 == 2)

score += dice2;

if (keepIndex\_2\_2 == 3)

score += dice3;

if (keepIndex\_2\_2 == 4)

score += dice4;

if (keepIndex\_2\_2 == 6)

score += dice6;

}

else if((keepIndex\_2\_2 == 5 && dice5 == 1) || (keepIndex\_2\_2 == 5 && dice5 == 4)){

if (keepIndex\_1\_2 == 1)

score += dice1;

if (keepIndex\_1\_2 == 2)

score += dice2;

if (keepIndex\_1\_2 == 3)

score += dice3;

if (keepIndex\_1\_2 == 4)

score += dice4;

if (keepIndex\_1\_2 == 6)

score += dice6;

}

if((keepIndex\_1\_2 == 6 && dice6 == 1) || (keepIndex\_1\_2 == 6 && dice6 == 4)){

if (keepIndex\_2\_2 == 1)

score += dice1;

if (keepIndex\_2\_2 == 2)

score += dice2;

if (keepIndex\_2\_2 == 3)

score += dice3;

if (keepIndex\_2\_2 == 4)

score += dice4;

if (keepIndex\_2\_2 == 5)

score += dice5;

}

else if((keepIndex\_2\_2 == 6 && dice6 == 1) || (keepIndex\_2\_2 == 6 && dice6 == 4)){

if (keepIndex\_1\_2 == 1)

score += dice1;

if (keepIndex\_1\_2 == 2)

score += dice2;

if (keepIndex\_1\_2 == 3)

score += dice3;

if (keepIndex\_2\_2 == 4)

score += dice4;

if (keepIndex\_1\_2 == 5)

score += dice5;

}

}

if(controller\_1 == 0 && controller\_4 == 0 || (controller\_1 == 1 && controller\_4 == 1)){ // the condition user didnt choose 1 and 4 or did choose 1 and 4

if(keepIndex\_1\_2 == 1 || keepIndex\_2\_2 == 1)

score += dice1;

if(keepIndex\_1\_2 == 2 || keepIndex\_2\_2 == 2)

score += dice2;

if(keepIndex\_1\_2 == 3 || keepIndex\_2\_2 == 3)

score += dice3;

if(keepIndex\_1\_2 == 4 || keepIndex\_2\_2 == 4)

score += dice4;

if(keepIndex\_1\_2 == 5 || keepIndex\_2\_2 == 5)

score += dice5;

if(keepIndex\_1\_2 == 6 || keepIndex\_2\_2 == 6)

score += dice6;

}

if (keepIndex\_1\_2 == 1 || keepIndex\_2\_2 == 1){ // this 6 if statments for updating the controllers

if (dice1 == 1)

controller\_1 = 1;

else if (dice1 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_2 == 2 || keepIndex\_2\_2 == 2){

if (dice2 == 1)

controller\_1 = 1;

else if (dice2 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_2 == 3 || keepIndex\_2\_2 == 3){

if (dice3 == 1)

controller\_1 = 1;

else if (dice3 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_2 == 4 || keepIndex\_2\_2 == 4){

if (dice4 == 1)

controller\_1 = 1;

else if (dice4 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_2 == 5 || keepIndex\_2\_2 == 5){

if (dice5 == 1)

controller\_1 = 1;

else if (dice5 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_2 == 6 || keepIndex\_2\_2 == 6){

if (dice6 == 1)

controller\_1 = 1;

else if (dice6 == 4)

controller\_4 = 1;

}

fflush(stdin);

printf("Shall I roll the remaining for you (Y/N)? ");

scanf("%c",&choice);

if (choice == 'N'){

if (controller\_1 == 1 && controller\_1 == 1)

return score;

else

return 0;

}

else if (choice == 'Y'){

dice1 = roll\_a\_dice();

dice2 = roll\_a\_dice();

dice3 = roll\_a\_dice();

dice4 = roll\_a\_dice();

dice5 = roll\_a\_dice();

dice6 = roll\_a\_dice();

if (keepIndex\_1 == 1 || keepIndex\_2 == 1 || keepIndex\_1\_2 == 1 || keepIndex\_2\_2 == 1) // making dice values 0 if previously used

dice1 = 0;

if (keepIndex\_1 == 2 || keepIndex\_2 == 2 || keepIndex\_1\_2 == 2 || keepIndex\_2\_2 == 2)

dice2 = 0;

if (keepIndex\_1 == 3 || keepIndex\_2 == 3 || keepIndex\_1\_2 == 3 || keepIndex\_2\_2 == 3)

dice3 = 0;

if (keepIndex\_1 == 4 || keepIndex\_2 == 4 || keepIndex\_1\_2 == 4 || keepIndex\_2\_2 == 4)

dice4 = 0;

if (keepIndex\_1 == 5 || keepIndex\_2 == 5 || keepIndex\_1\_2 == 5 || keepIndex\_2\_2 == 5)

dice5 = 0;

if (keepIndex\_1 == 6 || keepIndex\_2 == 6 || keepIndex\_1\_2 == 6 || keepIndex\_2\_2 == 6)

dice6 = 0;

printf("You got -> ");

if (dice1 != 0)

printf("[Dice 1]: %d ",dice1);

if (dice2 != 0)

printf("[Dice 2]: %d ",dice2);

if (dice3 != 0)

printf("[Dice 3]: %d ",dice3);

if (dice4 != 0)

printf("[Dice 4]: %d ",dice4);

if (dice5 != 0)

printf("[Dice 5]: %d ",dice5);

if (dice6 != 0)

printf("[Dice 6]: %d",dice6);

if(dice1 != 0){ // this 6 if statements for setting the last two indexes to keepIndex\_1\_3 and keepIndex\_2\_3

if (keepIndex\_1\_3 == 9)

keepIndex\_1\_3 = 1;

else

keepIndex\_2\_3 = 1;

}

if(dice2 != 0){

if (keepIndex\_1\_3 == 9)

keepIndex\_1\_3 = 2;

else

keepIndex\_2\_3 = 2;

}

if(dice3 != 0){

if (keepIndex\_1\_3 == 9)

keepIndex\_1\_3 = 3;

else

keepIndex\_2\_3 = 3;

}

if(dice4 != 0){

if (keepIndex\_1\_3 == 9)

keepIndex\_1\_3 = 4;

else

keepIndex\_2\_3 = 4;

}

if(dice5 != 0){

if (keepIndex\_1\_3 == 9)

keepIndex\_1\_3 = 5;

else

keepIndex\_2\_3 = 5;

}

if(dice6 != 0){

if (keepIndex\_1\_3 == 9)

keepIndex\_1\_3 = 6;

else

keepIndex\_2\_3 = 6;

}

if (((controller\_1 == 1) && (controller\_4 == 0)) || ((controller\_1 == 0) && (controller\_4 == 1))){ // if user just selects one dice which value's is 1 or 4

if((keepIndex\_1\_3 == 1 && dice1 == 1) || (keepIndex\_1\_3 == 1 && dice1 == 4)){

if (keepIndex\_2\_3 == 2)

score += dice2;

if (keepIndex\_2\_3 == 3)

score += dice3;

if (keepIndex\_2\_3 == 4)

score += dice4;

if (keepIndex\_2\_3 == 5)

score += dice5;

if (keepIndex\_2\_3 == 6)

score += dice6;

}

else if((keepIndex\_2\_3 == 1 && dice1 == 1) || (keepIndex\_2\_3 == 1 && dice1 == 4)){

if (keepIndex\_1\_3 == 2)

score += dice2;

if (keepIndex\_1\_3 == 3)

score += dice3;

if (keepIndex\_1\_3 == 4)

score += dice4;

if (keepIndex\_1\_3 == 5)

score += dice5;

if (keepIndex\_1\_3 == 6)

score += dice6;

}

if((keepIndex\_1\_3 == 2 && dice2 == 1) || (keepIndex\_1\_3 == 2 && dice2 == 4)){

if (keepIndex\_2\_3 == 1)

score += dice1;

if (keepIndex\_2\_3 == 3)

score += dice3;

if (keepIndex\_2\_3 == 4)

score += dice4;

if (keepIndex\_2\_3 == 5)

score += dice5;

if (keepIndex\_2\_3 == 6)

score += dice6;

}

else if((keepIndex\_2\_3 == 2 && dice2 == 1) || (keepIndex\_2\_3 == 2 && dice2 == 4)){

if (keepIndex\_1\_3 == 1)

score += dice1;

if (keepIndex\_1\_3 == 3)

score += dice3;

if (keepIndex\_1\_3 == 4)

score += dice4;

if (keepIndex\_1\_3 == 5)

score += dice5;

if (keepIndex\_1\_3 == 6)

score += dice6;

}

if((keepIndex\_1\_3 == 3 && dice3 == 1) || (keepIndex\_1\_3 == 3 && dice3 == 4)){

if (keepIndex\_2\_3 == 1)

score += dice1;

if (keepIndex\_2\_3 == 2)

score += dice2;

if (keepIndex\_2\_3 == 4)

score += dice4;

if (keepIndex\_2\_3 == 5)

score += dice5;

if (keepIndex\_2\_3 == 6)

score += dice6;

}

else if((keepIndex\_2\_3 == 3 && dice3 == 1) || (keepIndex\_2\_3 == 3 && dice3 == 4)){

if (keepIndex\_1\_3 == 1)

score += dice1;

if (keepIndex\_1\_3 == 2)

score += dice2;

if (keepIndex\_1\_3 == 4)

score += dice4;

if (keepIndex\_1\_3 == 5)

score += dice5;

if (keepIndex\_1\_3 == 6)

score += dice6;

}

if((keepIndex\_1\_3 == 4 && dice4 == 1) || (keepIndex\_1\_3 == 4 && dice4 == 4)){

if (keepIndex\_2\_3 == 1)

score += dice1;

if (keepIndex\_2\_3 == 2)

score += dice2;

if (keepIndex\_2\_3 == 3)

score += dice3;

if (keepIndex\_2\_3 == 5)

score += dice5;

if (keepIndex\_2\_3 == 6)

score += dice6;

}

else if((keepIndex\_2\_3 == 4 && dice4 == 1) || (keepIndex\_2\_3 == 4 && dice4 == 4)){

if (keepIndex\_1\_3 == 1)

score += dice1;

if (keepIndex\_1\_3 == 2)

score += dice2;

if (keepIndex\_1\_3 == 3)

score += dice3;

if (keepIndex\_1\_3 == 5)

score += dice5;

if (keepIndex\_1\_3 == 6)

score += dice6;

}

if((keepIndex\_1\_3 == 5 && dice5 == 1) || (keepIndex\_1\_3 == 5 && dice5 == 4)){

if (keepIndex\_2\_3 == 1)

score += dice1;

if (keepIndex\_2\_3 == 2)

score += dice2;

if (keepIndex\_2\_3 == 3)

score += dice3;

if (keepIndex\_2\_3 == 4)

score += dice4;

if (keepIndex\_2\_3 == 6)

score += dice6;

}

else if((keepIndex\_2\_3 == 5 && dice5 == 1) || (keepIndex\_2\_3 == 5 && dice5 == 4)){

if (keepIndex\_1\_3 == 1)

score += dice1;

if (keepIndex\_1\_3 == 2)

score += dice2;

if (keepIndex\_1\_3 == 3)

score += dice3;

if (keepIndex\_1\_3 == 4)

score += dice4;

if (keepIndex\_1\_3 == 6)

score += dice6;

}

if((keepIndex\_1\_3 == 6 && dice6 == 1) || (keepIndex\_1\_3 == 6 && dice6 == 4)){

if (keepIndex\_2\_3 == 1)

score += dice1;

if (keepIndex\_2\_3 == 2)

score += dice2;

if (keepIndex\_2\_3 == 3)

score += dice3;

if (keepIndex\_2\_3 == 4)

score += dice4;

if (keepIndex\_2\_3 == 5)

score += dice5;

}

else if((keepIndex\_2\_3 == 6 && dice6 == 1) || (keepIndex\_2\_3 == 6 && dice6 == 4)){

if (keepIndex\_1\_3 == 1)

score += dice1;

if (keepIndex\_1\_3 == 2)

score += dice2;

if (keepIndex\_1\_3 == 3)

score += dice3;

if (keepIndex\_1\_3 == 4)

score += dice4;

if (keepIndex\_1\_3 == 5)

score += dice5;

}

}

if((controller\_1 == 0 && controller\_4 == 0) || (controller\_1 == 1 && controller\_4 == 1)){ // the condition user didnt choose 1 and 4 or did choose 1 and 4

if(keepIndex\_1\_3 == 1 || keepIndex\_2\_3 == 1)

score += dice1;

if(keepIndex\_1\_3 == 2 || keepIndex\_2\_3 == 2)

score += dice2;

if(keepIndex\_1\_3 == 3 || keepIndex\_2\_3 == 3)

score += dice3;

if(keepIndex\_1\_3 == 4 || keepIndex\_2\_3 == 4)

score += dice4;

if(keepIndex\_1\_3 == 5 || keepIndex\_2\_3 == 5)

score += dice5;

if(keepIndex\_1\_3 == 6 || keepIndex\_2\_3 == 6)

score += dice6;

}

if (keepIndex\_1\_3 == 1 || keepIndex\_2\_3 == 1){ // this 6 if statements for updating the controllers

if (dice1 == 1)

controller\_1 = 1;

else if (dice1 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_3 == 2 || keepIndex\_2\_3 == 2){

if (dice2 == 1)

controller\_1 = 1;

else if (dice2 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_3 == 3 || keepIndex\_2\_3 == 3){

if (dice3 == 1)

controller\_1 = 1;

else if (dice3 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_3 == 4 || keepIndex\_2\_3 == 4){

if (dice4 == 1)

controller\_1 = 1;

else if (dice4 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_3 == 5 || keepIndex\_2\_3 == 5){

if (dice5 == 1)

controller\_1 = 1;

else if (dice5 == 4)

controller\_4 = 1;

}

if (keepIndex\_1\_3 == 6 || keepIndex\_2\_3 == 6){

if (dice6 == 1)

controller\_1 = 1;

else if (dice6 == 4)

controller\_4 = 1;

}

if(controller\_1 == 0 || controller\_4 == 0)

score = 0;

return score;

}

else

printf("\n\nPlease enter a valid choice!!!\n\n");

}

else

printf("\n\nPlease enter a valid choice!!!\n\n");

}

else

printf("\n\nPlease enter a valid choice!!!\n\n");

}

void scoresheet(int score\_comp, int score\_your){

printf("Our scoresheet:\n");

printf("=====================\n");

printf("My score \tYour score\n");

printf("%d \t\t%d\n\n", score\_comp, score\_your);

}