

```
#include<stdio.h>
#include<string.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
```

```
int diceroll();//function declaration
void display_welcome();
```

```
struct {
    char name[50];
    float freq[6];
    float average;
    float total;
    float percent[18];
    float sum[15];
    int freq1[6];
    int highest;
    int roll[3];
    int sum1;
    int count;
    } player[5];
```

```
int main()
{
```

```
FILE *fp;//to create .txt file
```

```
int a, b, c, num = 0, dice = 0, max, game=0;
int count = 0;
char stop = 'f';
```

```
srand(time(NULL));
system("clear");// to clear the screen
```

```
display_welcome();
```

```
if (num=1 || num<6){
    printf("\nEnter the number of players\n");
    scanf("%d",&num);
}
```

```
    getchar();
for ( a=1; a<=num; a++) {
    printf("\nPlease enter the name of Player %d : \n",a+0);
    player[a].count=0;
    gets(player[a].name);
}
```

```
system("clear");
```

```
if (dice=0 || dice > 3);
{
printf("Number of dices want to be used : ");
scanf("%d",&dice);
}
```

```
system("clear");
```

```
for (a=0;a<6;a++) {
for (c=0;c<18;c++) {
    player[a].sum[c]=0;
    }
}
```

```
while ( stop!='f') {
count++;
```

```
if ( dice==1 ) {
    for ( a=0; a<num; a++) {
        max = 1;
        printf("Player %d : %15s : \n",a+1,player[a].name);
```

```
        do {
```

```
            player[a].roll[0] = diceroll();
```

```
            if ( player[a].roll[0]==6 && max<3 ) {
                printf("\n\t\t\t\t\tYou rolled a 6.\n");
            } else
```

```
                switch (player[a].roll[0]) {
```

```
    case 1: player[a].freq[0]++;
        break;
    case 2: player[a].freq[1]++;
        break;
    case 3: player[a].freq[2]++;
        break;
    case 4: player[a].freq[3]++;
        break;
    case 5: player[a].freq[4]++;
        break;
    case 6: player[a].freq[5]++;
        break;
}
```

```
max++;
} while ( player[a].roll[0]==6 && max<4 );
```

```
player[a].sum1 = player[a].roll[0];
```

```
    for (c=0;c<6;c++) {
    if (player[a].sum1== c+1) {
    player[a].sum[c]++;
    }
    }
```

```
    printf("\n\t\t\t\t\tRoll %d : %d --> sum: %d\n",count,player[a].roll[0],player[a].sum1);
}
}
```

```
if ( dice==2 ) {
    for ( a=0; a<num; a++ ) {
        max = 1;
        printf("Player %d : %15s :\n",a+1,player[a].name);
```

```
        do {
```

```
            player[a].roll[0]=diceroll();
            player[a].roll[1]=diceroll();
```

```

if ( player[a].roll[0]==player[a].roll[1] && max<3 ) {
    printf("\n\t\t\t\t\tYou rolled a double: %d %d\n",player[a].roll[a],player[a].roll[a]);
}
else for ( b=0; b<2; b++ ) {
    switch (player[a].roll[b]) {
        case 1: player[a].freq[0]++;
            break;
        case 2: player[a].freq[1]++;
            break;
        case 3: player[a].freq[2]++;
            break;
        case 4: player[a].freq[3]++;
            break;
        case 5: player[a].freq[4]++;
            break;
        case 6: player[a].freq[5]++;
            break;
    }
}

```

```

max++;
} while ( player[a].roll[0]==player[a].roll[1] && max<4 );

```

```

player[a].sum1 = player[a].roll[0] + player[a].roll[1];

```

```

for (c=0;c<12;c++) {
    if (player[a].sum1 == c+1) {
        player[a].sum[c]++;
    }
}

```

```

    printf("\n\t\t\t\t\tRoll %d : %d %d -->
sum: %d\n",count,player[a].roll[0],player[a].roll[1],player[a].sum1);
}
}

```

```

if ( dice==3 ) {

```

```

for ( a=0; a<num; a++ ) {
max = 1;
printf("Player %d : %15s :\n",a+1,player[a].name);

do {

    player[a].roll[0] = diceroll();
    player[a].roll[1] = diceroll();
    player[a].roll[2] = diceroll();

    if (player[a].roll[0]==player[a].roll[1] && player[a].roll[1]==player[a].roll[2] && max<3) {
        printf("\n\t\t\t\t\tYou rolled a
triple: %d %d %d\n",player[a].roll[0],player[a].roll[1],player[a].roll[2]);
    }
    else for( b=0; b<3; b++ ) {
        switch (player[a].roll[b]) {
        case 1: player[a].freq[0]++;
            break;
        case 2: player[a].freq[1]++;
            break;
        case 3: player[a].freq[2]++;
            break;
        case 4: player[a].freq[3]++;
            break;
        case 5: player[a].freq[4]++;
            break;
        case 6: player[a].freq[5]++;
            break;
        }
    }

max++;
} while
( player[a].roll[0]==player[a].roll[1] && player[a].roll[1]==player[a].roll[2] && max<4 );

player[a].sum1 = player[a].roll[0] + player[a].roll[1] + player[a].roll[2];

for (c=0;c<18;c++) {
if (player[a].sum1 == c+1) {
player[a].sum[c]++;
}
}

```

```

}

printf("\n\t\t\t\t\tRoll %d : %d %d %d -->
sum: %d\n",count,player[a].roll[0],player[a].roll[1],player[a].roll[2],player[a].sum1);
}
}

for ( a=0; a<num; a++ ) {
player[a].total = player[a].total + player[a].sum1;
}

getchar();
printf("\n\n\nEnter any character to continue playing, 'f' to end game > ");
scanf("%c",&stop);
system("clear");

/*for( a=0; a<num; a++ ) {
printf("Player %d : %s\n",a+1,player[a].name);

player[a].highest=0;

for( b=0; b<6; b++ ) {
// printf("Dice %d : %.0f \n",b+1,player[a].freq[b]);
if ( player[a].freq[b] >= player[a].highest ) {

player[a].mostfreq[b] = b+1;

if ( player[a].freq[b] > player[a].highest ) {
player[a].highest = player[a].freq[b];
} else player[a].highest = player[a].freq[b];
}
}
}

printf("\n\n %d rolls, Player %s most frequently roll",count,player[a].name);
for ( b=0; b<6; b++ ) {

```

```

if (player[a].freq[b]==player[a].highest) {
    printf(" '%d' ",player[a].mostfreq[b]);
}
}
printf(" on %d dices",player[a].highest);

player[a].avg = player[a].total/count;
printf("\n\tAverage roll : %.2f\n",player[a].avg);

for( c=0; c<18; c++ ) {
    player[a].per[c] = (player[a].sum[c]/count)*100;
    printf("\t\tRoll %d : \t %.2f%% of the time on %.0f
dices\n",x+1,player[a].per[c],player[a].sum[c]);
}

*/

do {
fp=fopen("a.txt","a");
} while (fp==NULL);

game++;
fprintf(fp,"Game %d :\n\n",game);

for( a=0; a<num; a++ ) {
    fprintf(fp,"\nPlayer %d : %s\n",a+1,player[a].name);

    player[a].highest=0;

    for( b=0; b<6; b++ ) {
        if ( player[a].freq[b] >= player[a].highest ) {

            player[a].freq[b] = b+1;

            if ( player[a].freq[b] > player[a].highest ) {
                player[a].highest = player[a].freq[b];
            } else player[a].highest = player[a].freq[b];
        }
    }
}

```

```

    }
}

fprintf(fp, "\n\n%d rolls, Player %s most frequently roll", count, player[a].name);
for ( b=0; b<6; b++ ) {
if (player[a].freq[b]==player[a].highest) {
    fprintf(fp, " '%d' ", player[a].freq1[b]);
    (fp, " on %d dices", player[a].highest);

player[a].average = player[a].total/count;
fprintf(fp, "\n\tAverage roll : %.2f\n", player[a].average);

    for( c=0; c<18; c++ ) {
        player[a].percent[c] = (player[a].sum[c]/count)*100;
        fprintf(fp, "\t\tRoll %d : \t %.2f%% of the time on %.0f
dices\n", c+1, player[a].percent[c], player[a].sum[c]);
    }
printf("\n\n\n");
}
}
}
}
return 0;
}

int diceroll()//function call
{

    int result;
    result = ((int) rand()%6) + 1;
    return (result);
}

void display_welcome(){
printf("Welcome!!!");
printf("\n\t\Dice Game of");
printf("\n\n\t\t\t\t\Naren & Cx");

```



```
return;  
}
```