Data structures assignment

Name: Vijay

Roll no:2019115120

---

1)

Code

```
// To get and display the players details
#include<stdio.h>
#include<stdlib.h>

struct players
{
    int player_id;
    char *name;
    struct score
    {
        int boundaries;
        int sixer;
    } score;
    int points;
    char *ratings;
};
typedef struct players players;
players num_of_players[11];
int size = sizeof(num_of_players)/sizeof(num_of_players[0]);

int comparator(const void* p, const void* q)
{
    const players *p1 = p;
    const players *p2 = q;
    return p2->points - p1->points;
}

void score_board()
{
    qsort(num_of_players, size,sizeof(players),comparator);
}

void ratings(int i)
{
```

```
        num_of_players[i].ratings = (struct player *)malloc(sizeof(char)* 10 )
;
        if (num_of_players[i].points >= 100)
            num_of_players[i].ratings = "Excellent";
        else if(num_of_players[i].points >= 50 && num_of_players[i].points < 1
00 )
            num_of_players[i].ratings = "Good";
        else
            num_of_players[i].ratings = "Fair";



}

void get(int number)
{


    for (int i = 0; i < number; i++)
    {
        num_of_players[i].name = (struct players *)malloc(sizeof(char)*15);

        printf("-------------------- for player %d --------------------
\n",i+1);
        printf("\nPlayer_id\t");
        scanf("%d",&num_of_players[i].player_id);
        printf("\nName\t");
        scanf("%s",num_of_players[i].name);
        printf("\nNo of boundaries\t");
        scanf("%d",&num_of_players[i].score.boundaries);
        printf("\nNo of sixer\t");
        scanf("%d",&num_of_players[i].score.sixer);
        num_of_players[i].points = (num_of_players[i].score.boundaries * 4) +
(num_of_players[i].score.sixer * 6);
        ratings(i);
    }

}

void display(int number)
{

    printf("-----------------------------------------------------------------
------------\n");
    printf("|%-5s||%-10s||%-15s||%-11s||%-6s||%-7s||%-
10s|\n","S no","player id","Name","Boundaries","sixer","Points","Ratings");
    printf("-----------------------------------------------------------------
------------\n");
```

```
    for (int i = 0; i < number; i++)
    {
        printf("|%-5d|",i+ 1);
        printf("|%-10d|",num_of_players[i].player_id);
        printf("|%-15s|",num_of_players[i].name);
        printf("|%-11d|",num_of_players[i].score.boundaries);
        printf("|%-6d|",num_of_players[i].score.sixer);
        printf("|%-7d|",num_of_players[i].points);
        printf("|%-10s|",num_of_players[i].ratings);
        printf("\n");
    }
    printf("-----------------------------------------------------------------------
------------\n");


}

int main()
{
    int no_of_players;
    printf("How many players do you want to enter\n");
    scanf("%d",&no_of_players);
    get(no_of_players);
    score_board();
    display(no_of_players);
    return 0;

}
```

Output

```
-------------------------------------------------------------------------------
|S no ||player id ||Name           ||Boundaries ||sixer ||Points ||Ratings   |
-------------------------------------------------------------------------------
|1    ||3945      ||VIGNESH        ||20         ||12    ||152    ||Excellent |
|2    ||4954      ||IRFAN          ||10         ||10    ||100    ||Excellent |
|3    ||3567      ||YOGESH         ||9          ||8     ||84     ||Good      |
|4    ||2045      ||AKILAN         ||10         ||2     ||52     ||Good      |
|5    ||3842      ||SIVA_MURUGAN   ||5          ||5     ||50     ||Good      |
|6    ||3953      ||SAM            ||3          ||6     ||48     ||Fair      |
|7    ||3         ||RAMYA          ||3          ||6     ||48     ||Fair      |
|8    ||3043      ||VIJAY          ||3          ||5     ||42     ||Fair      |
|9    ||4355      ||OSLO           ||3          ||5     ||42     ||Fair      |
|10   ||3456      ||RAMANI         ||3          ||5     ||42     ||Fair      |
|11   ||4933      ||HARI           ||3          ||5     ||42     ||Fair      |
-------------------------------------------------------------------------------
```

2)

```
#include <stdio.h>
#include <stdlib.h>
int StringToInt(char *st)
{
    int strToNum=0,i=0;
    while(*(st+i)!='\0')
    {
        strToNum=strToNum*10+st[i++]-'0';
    }
    return strToNum;
}
int sumOfDigits(char *st)
{
    int num=StringToInt(st);
    int sum=0;
    while(num>0)
    {
        sum+=num%10;
        num/=10;
    }
    return sum;
}
char* IntegerToString(int n)
{
    char *st=(char *)malloc(sizeof(char)*9);
    int rev=0,p=n,c=0,i;
    while(n>0)
    {
        rev=rev*10+n%10;
        n/=10;
        c++;
    }
    for(i=0;i<c;i++)
    {
        *(st+i)=rev%10+'0';
        rev/=10;
    }
    *(st+i)='\0';
    return st;
}
void main()
{
    char *str=(char *)malloc(sizeof(char)*7);
    char *str1=(char *)malloc(sizeof(char)*7);
    printf("Enter the first number (The entered number is considered to be a s
tring) : ");
```

```
    gets(str);
    int num1=StringToInt(str);
    printf("The integer value of the string : %d\n",num1);
    printf("The sum of the digits in %s = %d\n",str,sumOfDigits(str));
    printf("Enter the second number (The entered number is considered to be a
string) : ");
    gets(str1);
    int num2=StringToInt(str1);
    printf("Integer sum of %d and %d = %s (The final result is a string) ",num
1,num2,IntegerToString(num1+num2));
}
```

Output

```
vijaysaravanan@VIJAY-PC:/mnt/c/Users/VIJAY/Vscode/Assignment$ ./a.out
Enter the first number (The entered number is considered to be a string) : 35533
The integer value of the string : 35533
The sum of the digits in 35533 = 19
Enter the second number (The entered number is considered to be a string) : 44553
Integer sum of 35533 and 44553 = 80086 (The final result is a string)
 vijaysaravanan@VIJAY-PC:/mnt/c/Users/VIJAY/Vscode/Assignment$
```