PPL Practice Problems

Name: Himani Verma

Admission No: U19CS075

1. **Write a program that requests the user to enter two integers. The program should then calculate and report the sum of all the integers between and including the two integers. At this point, assume that the smaller integer is entered first. For example, if the user enters 2 and 9, the program should report that the sum of all the integers from 2 through 9 is 44.**

Source Code:

```
#include <iostream>
using namespace std;

int WinMain(){
    int a,b;
    long sum = 0;
    cout << "Enter two integer : ";
    cin >> a >> b;
    if(a>b){
        swap(a,b);
    }
    for(int i = a;i<=b;i++){
        sum+=i;
    }
    cout << "Sum of numbers between " << a << " and " << b << " including both
numbers is : " << sum << endl;
    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                  Code

PS D:\documents\PPL\Practice Questions> cd "d:\documents\PPL\Practice Questions\" ; if ($?) { g++ 1.cpp -o 1 } ; if ($?) { .\1 }
Enter two integer : 3 5
Sum of numbers between 3 and 5 including both numbers is : 12
PS D:\documents\PPL\Practice Questions> cd "d:\documents\PPL\Practice Questions\" ; if ($?) { g++ 1.cpp -o 1 } ; if ($?) { .\1 }
Enter two integer : 4 10
Sum of numbers between 4 and 10 including both numbers is : 49
PS D:\documents\PPL\Practice Questions>
```

2. **Write a program that opens a text file, reads it character-by-character to the end of the file,and reports the number of characters in the file.**

File.txt

```
ggurgugfrhgo
nfiurh
fiehf
feiuhfurhfuie
ggi
```

Source Code:

```
#include<iostream>
#include<fstream>
using namespace std;

int WinMain(){
```

```
    char ch;
    ifstream myFile;
    int count = 0;
    myFile.open("file.txt",ios::in);
    while(!myFile.eof()){
        myFile.get(ch);
        count++;
    }
    myFile.close();
    cout<<"Total characters in the file are: "<<count<<endl;
    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                   Code

PS D:\documents\PPL\Practice Questions> cd "d:\documents\PPL\Practice Questions\" ; if ($?) { g++ 2.cpp -o 2 } ; if ($?) { .\2 }
Total characters in the file are: 52
PS D:\documents\PPL\Practice Questions>
```

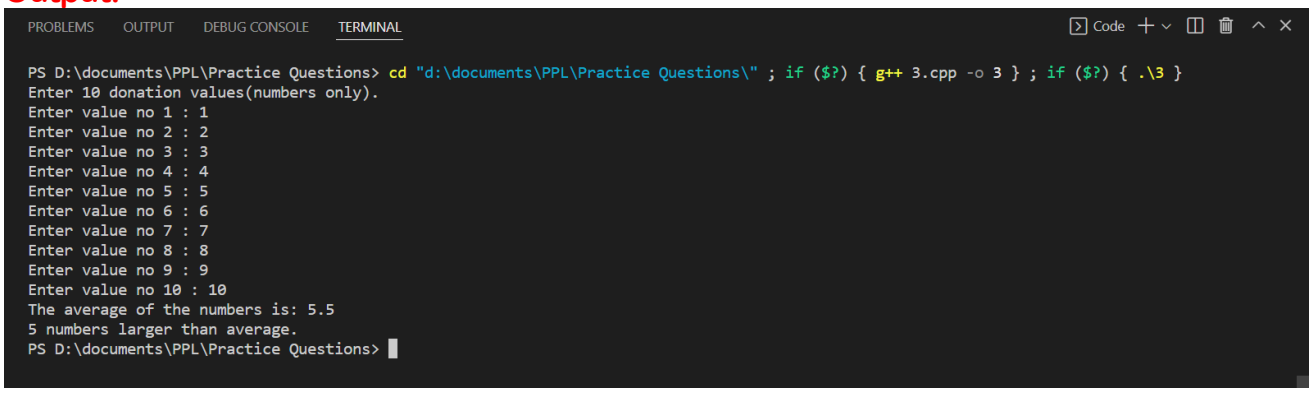
3. **Write a program that reads up to 10 donation values into an array of double. (Or, if you prefer, use an array template object.) The program should terminate input on non-numeric input. It should report the average of the numbers and also report how many numbers in the array are larger than the average.**

Source Code:

```
#include <iostream>
using namespace std;

int main()
{
    double total;
    int bigger=0;
    int values=0;
    double donation[10];

    cout<<"Enter 10 donation values(numbers only). "<<endl;
    cout<<"Enter value no 1 : ";

    while (values<10)
    {
        cin>>donation[values];
        if(!cin){
            cout<<"Kindly enter a numeric value!"<<endl;
            exit(0);
        }
        total+=donation[values];
        values++;
        if (values<10){
            cout<<"Enter value no " <<(values+1)<<" : ";
        }
    }

    double average = total/(double)10;
```

```
    for ( values=0; values<10; values++)
    {
        if (donation[values]> average)
        {
            bigger++;
        }

    }
    cout<<"The average of the numbers is: "<<average<<endl;
    cout<< bigger <<" numbers larger than average. \n";
    return 0;
}
```

Output:



4. **Here is a structure declaration:**

```
struct box{
        char maker[40];
        float height;
        float width;
        float length;
        float volume;
};
```

a. **Write a function that passes a box structure by value and that displays the value of each member.**

b. **Write a function that passes the address of a box structure and that sets the volume member to the product of the other three dimensions.**

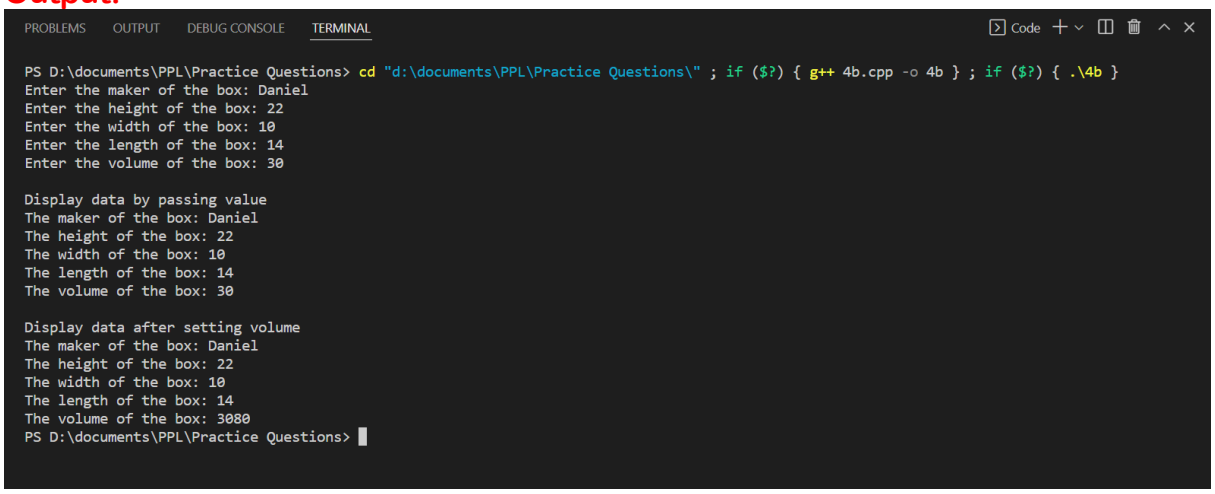
c. **Write a simple program that uses these two functions.**

Source Code: (for a, b & c)

```
#include<iostream>
using namespace std;

struct box
{
    char maker[40];
    float height;
    float width;
    float length;
    float volume;
};
```

```cpp
void displayValue(box b){
    cout<<"The maker of the box: "<<b.maker<<endl;
    cout<<"The height of the box: "<<b.height<<endl;
    cout<<"The width of the box: "<<b.width<<endl;
    cout<<"The length of the box: "<<b.length<<endl;
    cout<<"The volume of the box: "<<b.volume<<endl;
}

void setVolume(box *b){
    b->volume = b->height*b->width*b->length;
}

int main(){
    box b1;
    cout<<"Enter the maker of the box: ";
    cin>>b1.maker;
    cout<<"Enter the height of the box: ";
    cin>>b1.height;
    cout<<"Enter the width of the box: ";
    cin>>b1.width;
    cout<<"Enter the length of the box: ";
    cin>>b1.length;
    cout<<"Enter the volume of the box: ";
    cin>>b1.volume;

    cout<<endl;
    cout<<"Display data by passing value"<<endl;
    displayValue(b1);

    setVolume(&b1);

    cout<<"\nDisplay data after setting volume"<<endl;
    displayValue(b1);
    return 0;
}
```

Output:



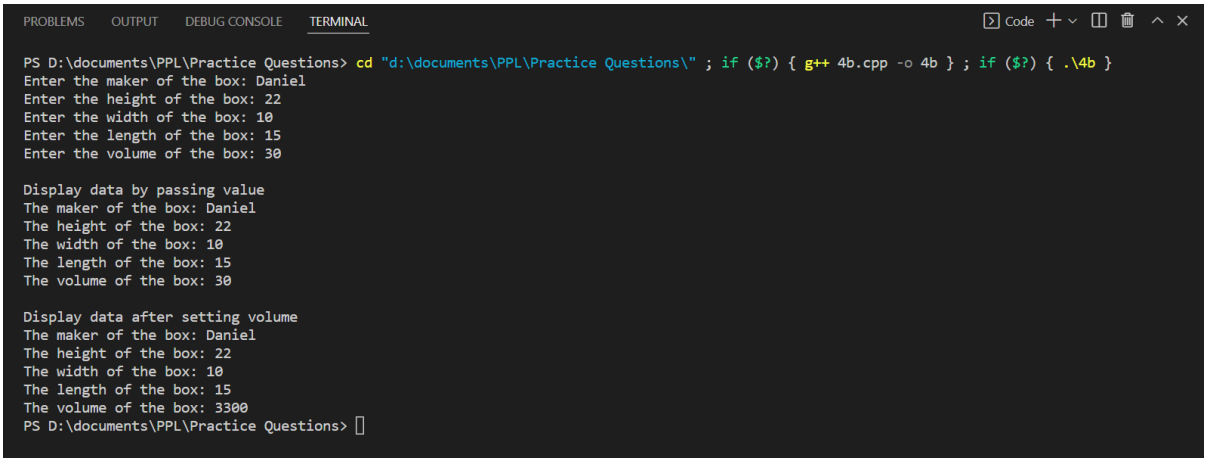
d. Write a function that has a reference to a box structure as its formal argument and displays the value of each member (Separate program d & e).

e. Write a function that has a reference to a box structure as its formal argument and sets the volume member to product of the other 3 dimensions.

Source Code:

```
#include<iostream>
using namespace std;

struct box
{
    char maker[40];
    float height;
    float width;
    float length;
    float volume;
};

void displayValue(box &b){
    cout<<"The maker of the box: "<<b.maker<<endl;
    cout<<"The height of the box: "<<b.height<<endl;
    cout<<"The width of the box: "<<b.width<<endl;
    cout<<"The length of the box: "<<b.length<<endl;
    cout<<"The volume of the box: "<<b.volume<<endl;
}

void setVolume(box &b){
    b.volume = b.height*b.width*b.length;
}

int main(){
    box b1;
    cout<<"Enter the maker of the box: ";
    cin>>b1.maker;
    cout<<"Enter the height of the box: ";
    cin>>b1.height;
    cout<<"Enter the width of the box: ";
    cin>>b1.width;
    cout<<"Enter the length of the box: ";
    cin>>b1.length;
    cout<<"Enter the volume of the box: ";
    cin>>b1.volume;

    cout<<endl;
    cout<<"Display data by passing value"<<endl;
    displayValue(b1);

    setVolume(b1);

    cout<<"\nDisplay data after setting volume"<<endl;
    displayValue(b1);
    return 0;
}
```

Output:



5. **Write a program that uses the following functions:**
   Fill_array() takes as arguments the name of an array of double values and an array size. It prompts the user to enter double values to be entered in the array. It ceases taking input when the array is full or when the user enters non-numeric input,and it returns the actual number of entries.
   Show_array() takes as arguments the name of an array of double values and an array size and displays the contents of the array.
   Reverse_array() takes as arguments the name of an array of double values and an array size and reverses the order of the values stored in the array.
   The program should use these functions to fill an array, show the array, reverse the array, show the array, reverse all but the first and last elements of the array,and then show the array.

   Source Code:

```cpp
#include <iostream>
using namespace std;
const int Max = 5;

int fill_array(double ar[], int limit);
void show_array(const double ar[], int n);
void reverse_array(double ar[], int n);

int main(){
    double properties[Max];
    int size = fill_array(properties, Max);
    cout << endl;
    cout<<"---------Show---------"<<endl;
    show_array(properties, size);
    cout << endl;
    cout<<"---------Reverse all---------"<<endl;
    reverse_array(properties, size);
    show_array(properties, size);
    cout << endl;
    cout<<"---------Reverse all but the first and last element---------"<<endl;
    reverse_array(properties + 1, size -2);
    show_array(properties, size);
    return 0;
 }

int fill_array(double ar[], int limit){
```

```
 double temp;
 int i;
 for(i = 0; i < limit; i++){
    cout << "Enter value " << (i + 1) << ": ";
    cin >> temp;
    if(!cin){
        cin.clear();
        while(cin.get() != '\n') continue;
        cout << "Bad input; input process terminated" << endl;
        break;
    }
    else if(temp < 0) break;
    ar[i] = temp;
    }
    return i;
 }

void show_array(const double ar[], int n){
 for (int i = 0; i < n; i++){
    cout << "Property " << (i + 1) << ": ";
    cout << ar[i] << endl;
    }
 }

void reverse_array(double ar[], int n){
    double temp;
    for(int i = 0; i < n/2; i++){
        temp = ar[i];
        ar[i] = ar[n - i - 1];
        ar[n - i - 1] = temp;
    }
 }
```

Output: