

Lab Assignment 7

Yeeshukant Singh | 200002082

Q1. Create a class called Time that maintains the time in 11:59:59 format. Include appropriate data members, constructors, and appropriate member functions in the class to set the time, update the time, etc.

Ans:

```
#include<iostream>
using namespace std;
class Time{
    int hour, min, sec;
public:
    Time(){}
    void set(int a, int b, int c){
        hour=a; min=b; sec=c;
    }
    void update(int a, int b, int c){
        if(sec+c>=60)
            b++;
        if(min+b>=60)
            a++;
        hour=(hour+a)%24;
        min=(min+b)%60;
        sec=(sec+c)%60;
        printf("Updated Time is: %02d:%02d:%02d\n", hour, min, sec);
    }
};

int main(){
    Time t;
    int hour, min, sec;
    printf("Set Time: ");
    cin>>hour>>min>>sec;
    if(hour>24||hour<0){
        cout<<"Time not valid\n";
        return 0;
    }
    else if(hour==24&&min>0){
        cout<<"Time not valid\n";
        return 0;
    }
    else if(min<0||min>59||sec<0||sec>59){
        cout<<"Time not valid\n";
        return 0;
    }
    t.set(hour, min, sec);
}
```

Lab Assignment 7

Yeeshukant Singh | 200002082

```
cout<<"Enter time update: ";
cin>>hour>>min>>sec;
t.update(hour, min,sec);
return 0;
}
```

Output:

```
~/My-files $ ./a.out
Set Time: 23 59 59
Enter time update: 0 0 1
Updated Time is: 00:00:00
~/My-files $ ./a.out
Set Time: 15 20 53
Enter time update: 0 30 34
Updated Time is: 15:51:27
```

Q2. Create a class called Fraction with the following characteristics. Member data is the fraction's numerator and denominator. Member functions should accept input from the user in the form 3/5, and output the fraction's value in the same format. Another member function should add two fraction values. Write a main() program that allows the user to repeatedly input two fractions and then displays their sum. After each operation, ask whether the user wants to continue.

Ans:

```
#include<iostream>
#include<algorithm>
using namespace std;
class Fraction{
    int nume, deno;
    char dummy;
public:
    void input(){
        cin>>nume>>dummy>>deno;
    }
    Fraction sum(Fraction f2){
        Fraction sum;
        int lcm=(deno*f2.deno)/__gcd(deno,f2.deno);
        sum.nume=(nume)*(lcm/deno)+(f2.nume)*(lcm/f2.deno);
        sum.deno=lcm;
        return sum;
    }
    void display(){
        printf("The sum is: %d/%d\n",nume,deno);
    }
}
```

Lab Assignment 7

Yeeshukant Singh | 200002082

```
};  
int main(){  
    Fraction f1;  
    Fraction f2;  
    Fraction SUM;  
    char yn;  
    do{  
        f1.input();  
        f2.input();  
        SUM=f1.sum(f2);  
        SUM.display();  
        cout<<"Do you want to continue (y/n):";  
        cin>>yn;  
    }while(yn=='y');  
    return 0;  
}
```

Output:

```
~/My-files $ ./a.out  
2/3 4/5  
The sum is: 22/15  
Do you want to continue (y/n):y  
1/2 1/3  
The sum is: 5/6  
Do you want to continue (y/n):n
```

Q3. Imagine a tollbooth at a bridge. Cars passing by the booth are expected to pay a 50 rupees toll. Mostly they do, but sometimes a car goes by without paying. The tollbooth keeps track of the number of cars that have gone by, and of the total amount of money collected. Model this tollbooth with a class called tollBooth. The two data items are a type int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both of these to 0. A member function called payingCar() increments the car total and adds 0.50 to the cash total. Another function, called nopayCar(), increments the car total but adds nothing to the cash total. Finally, a member function called display() displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car.

Ans:

```
#include<iostream>  
using namespace std;  
class tollBooth{  
    int cnt;  
    double cash;
```

Lab Assignment 7

Yeeshukant Singh | 200002082

```
public:
tollBooth(){
    cnt=0;
    cash=0;
}
void paying(){
    cnt++;
    cash=cash+0.50;
}
void nonpaying(){
    cnt++;
}
void display(){
    printf("Car count: %d Cash count: %0.2lf\n",cnt,cash);
}
};

int main(){
    tollBooth t1;
    char yn;
    cout<<"paying(1),nonpaying(0),exit(n): ";
    cin>>yn;
    while(yn=='1' || yn=='0'){
        if(yn=='1')
            t1.paying();
        else if(yn=='0')
            t1.nonpaying();
        cout<<"paying(1),nonpaying(0),exit(n): ";
        cin>>yn;
        if(yn!='0' || yn!='1' || yn!='n')
            cout<<"Invalid input\n";
    }
    t1.display();
    return 0;
}
```

Output:

```
~/My-files $ ./a.out
paying(1),nonpaying(0),exit(n): 1
paying(1),nonpaying(0),exit(n): 1
paying(1),nonpaying(0),exit(n): 0
paying(1),nonpaying(0),exit(n): 1
paying(1),nonpaying(0),exit(n): 0
```

Lab Assignment 7

Yeeshukant Singh | 200002082

```
paying(1),nonpaying(0),exit(n): n  
Car count: 5 Cash count: 1.50
```