

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM



MÔN LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG
BÀI TẬP THỰC HÀNH 1

GVHD: Nguyễn Ngọc Quý

Sinh viên thực hiện: Nguyễn Thanh Tùng

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

(Ký tên và ghi rõ họ tên)

Mục Lục

1. Bài tập 1: Xây dựng lớp ngày tháng năm bao gồm:	6
1.1 Phương thức của class NgayThangNam:	7
1.1.1 - Phương thức Nhap():	7
1.1.2 - Phương thức Xuat():	8
1.1.3 - Phương thức NgayThangNamTiepTheo():	8
2. Bài tập 2: Xây dựng lớp phân số bao gồm:	14
2.1 Phương thức của class PhanSo:	16
2.1.1 - Phương thức Nhap():	16
2.1.2 - Phương thức Xuat():	16
2.1.3 - Phương thức RutGon():	16
2.1.4 - Phương thức Tong():	17
2.1.5 - Phương thức Hieu():	19
2.1.6 - Phương thức Tich():	21
2.1.7 - Phương thức Thuong():	23
2.1.8 - Phương thức Sosanh():	25
3. Bài tập 3: Xây dựng lớp số phức bao gồm:	26
3.1 Phương thức của class SoPhuc	28
3.1.1 - Phương thức Nhap():	28
3.1.2 - Phương thức Xuat():	29
3.1.3 - Phương thức Tong():	29
3.1.4 - Phương thức Hieu():	30
3.1.5 - Phương thức Tich():	30
3.1.6 - Phương thức Thuong():	31
4. Bài tập 4: Xây dựng lớp giờ phút giây bao gồm:	32
4.1 Phương thức của class GioPhutGiay	34
4.1.1 - Phương thức Nhap():	34
4.1.2 - Phương thức Xuat():	34
4.1.3 - Phương thức TinhCongThemMotGiay():	35

DANH MỤC BẢNG

Bảng 1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp NgayThangNam	7
Bảng 2. Nội dung của phương thức Nhap()	8
Bảng 3. Nội dung của phương thức Xuat()	8
Bảng 4. Nội dung của hướng giải quyết tháng 2 trong hàm NgayThangNamTiepTheo()	10
Bảng 5. Nội dung của hướng giải quyết tháng có 30 ngày trong hàm NgayThangNamTiepTheo()	11
Bảng 6. Bảng 1.5. Nội dung của hướng giải quyết tháng có 31 ngày trong hàm NgayThangNamTiepTheo()	13
Bảng 7. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo	15
Bảng 8. Nội dung của phương thức Nhap() của lớp PhanSo	16
Bảng 9. Nội dung của phương thức Xuat() của lớp PhanSo	16
Bảng 10. Nội dung của hàm RutGon() của lớp PhanSo()	17
Bảng 11. Nội dung của phương thức Tong() của lớp PhanSo	19
Bảng 12. Nội dung của phương thức Hieu() của lớp PhanSo	21
Bảng 13. Nội dung của phương thức Tich() của lớp PhanSo	23
Bảng 14. Nội dung của phương thức Thuong() của lớp PhanSo	25
Bảng 15. Nội dung của phương thức Sosanh() của lớp PhanSo	26
Bảng 16. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo	28
Bảng 17. Nội dung của phương thức Nhap() của lớp SoPhuc	28
Bảng 18. Nội dung của phương thức Xuat() của lớp SoPhuc	29
Bảng 19. Nội dung của phương thức Tong() trong lớp SoPhuc	30
Bảng 20. Nội dung của phương thức Hieu() trong lớp SoPhuc	30
Bảng 21. Nội dung của phương thức Tich() trong lớp SoPhuc	31
Bảng 22. Nội dung của phương thức Thuong() trong lớp SoPhuc	32
Bảng 23. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp GioPhutGiay	34
Bảng 24. Nội dung của phương thức Nhap() của lớp GioPhutGiay	34

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Bảng 25. Nội dung của phương thức Xuất() của lớp GioPhutGiay	35
Bảng 26. Nội dung của phương thức TinhCongThemMotGiay() của lớp GioPhutGiay	38

DANH MỤC HÌNH ẢNH

Hình 1. Class diagram của lớp NgayThangNam.....	6
____Hình 1.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp NgayThangNam	7
____Hình 1.2. Nội dung của phương thức Nhập().....	8
____Hình 1.3. Nội dung của phương thức Xuất().....	8
____Hình 1.4. Nội dung của hướng giải quyết tháng 2 trong hàm NgayThangNamTiepTheo().....	9
____Hình 1.5. Nội dung của hướng giải quyết tháng có 30 ngày trong hàm NgayThangNamTiepTheo().....	11
____Hình 1.6. Nội dung của hướng giải quyết tháng có 31 ngày trong hàm NgayThangNamTiepTheo().....	12
Hình 2. Class diagram của lớp PhanSo.....	14
____Hình 2.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo	15
____Hình 2.2. Nội dung của phương thức Nhập() của lớp PhanSo.....	16
____Hình 2.3. Nội dung của phương thức Xuất() của lớp PhanSo	16
____Hình 2.4. Nội dung của hàm RutGon() của lớp PhanSo()	17
____Hình 2.5. Nội dung của phương thức Tong() của lớp PhanSo.....	18
____Hình 2.6. Nội dung của phương thức Hieu() của lớp PhanSo	20
____Hình 2.7. Nội dung của phương thức Tich() của lớp PhanSo.....	22
____Hình 2.8. Nội dung của phương thức Thuong() của lớp PhanSo.....	24
____Hình 2.9. Nội dung của phương thức Sosanh() của lớp PhanSo.....	25
Hình 3. Class diagram của lớp SoPhuc.....	27
____Hình 3.2. Nội dung của phương thức Nhập() của lớp SoPhuc.....	28
____Hình 3.3. Nội dung của phương thức Xuất() của lớp SoPhuc	29
____Hình 3.4. Nội dung của phương thức Tong() trong lớp SoPhuc	29

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

____	Hình 3.5. Nội dung của phương thức Hieu() trong lớp SoPhuc.....	30
____	Hình 3.6. Nội dung của phương thức Tich() trong lớp SoPhuc	31
____	Hình 3.7. Nội dung của phương thức Thuong() trong lớp SoPhuc	32
	Hình 4. Class diagram của lớp GioPhutGiay	33
____	Hình 4.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp GioPhutGiay	33
____	Hình 4.2. Nội dung của phương thức Nhap() của lớp GioPhutGiay	34
____	Hình 4.3. Nội dung của phương thức Xuat() của lớp GioPhutGiay	34
____	Hình 4.4. Nội dung của phương thức TinhCongThemMotGiay() của lớp GioPhutGiay	36

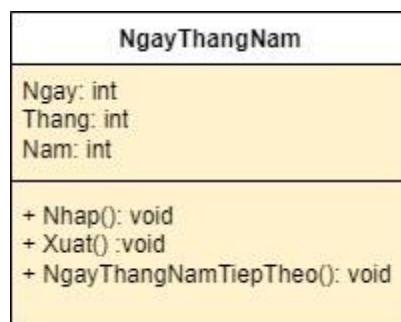
NỘI DUNG BÀI LÀM

1. Bài tập 1: Xây dựng lớp ngày tháng năm bao gồm:

- Thuộc tính: iNgay, iThang, iNam
- Phương thức: Nhap(), Xuat(), NgayThangNamTiepTheo()

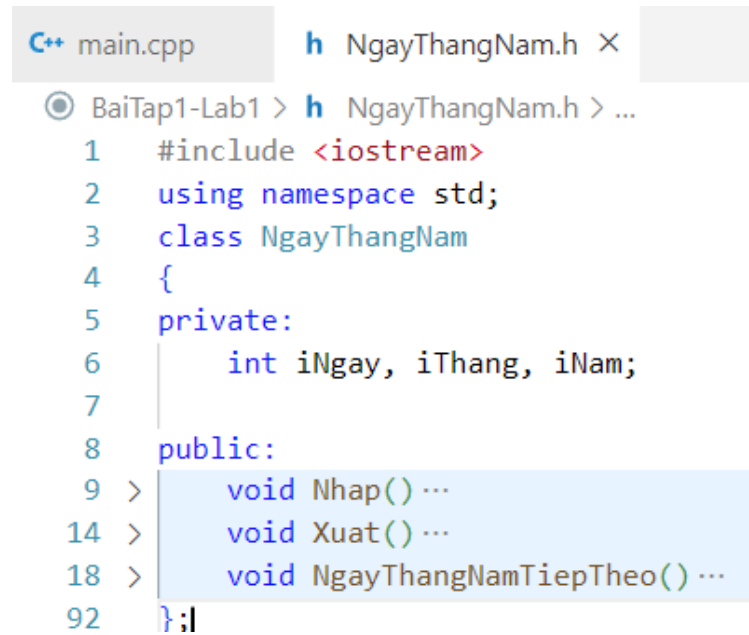
Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính, phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức trong hàm main()

Class diagram của lớp NgayThangNam:



Hình 1. Class diagram của lớp NgayThangNam

Thực hiện xây dựng lớp, khai báo thuộc tính, phương thức của lớp NgayThangNam.



```
C++ main.cpp h NgayThangNam.h X
BaiTap1-Lab1 > h NgayThangNam.h > ...
1  #include <iostream>
2  using namespace std;
3  class NgayThangNam
4  {
5  private:
6      int iNgay, iThang, iNam;
7
8  public:
9  >     void Nhap() ...
14 >     void Xuat() ...
18 >     void NgayThangNamTiepTheo() ...
92 };|
```

Hình 1.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp NgayThangNam

```
#include <iostream>
using namespace std;
class NgayThangNam
{
private:
    int iNgay, iThang, iNam;

public:
    void Nhap()...
    void Xuat()...
    void NgayThangNamTiepTheo()...
};
```

Bảng 1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp NgayThangNam

1.1 Phương thức của class NgayThangNam:

1.1.1 - Phương thức Nhap():

- Nội dung: Nhập vào ngày tháng năm từ bàn phím

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
9      void Nhap()  
10     {  
11         cout << "Nhap vao ngay thang nam: ";  
12         cin >> iNgay >> iThang >> iNam;  
13     }
```

Hình 1.2. Nội dung của phương thức Nhap()

```
void Nhap()  
{  
    cout << "Nhap vao ngay thang nam: ";  
    cin >> iNgay >> iThang >> iNam;  
}
```

Bảng 2. Nội dung của phương thức Nhap()

1.1.2 - Phương thức Xuat():

- Nội dung: Xuất ra ngày tháng năm vừa nhập

```
14     void Xuat()  
15     {  
16         cout << "Ngày tháng năm vừa nhập là: " << iNgay << "/" << iThang <<  
17         "/" << iNam << endl;  
18     }
```

Hình 1.3. Nội dung của phương thức Xuat()

```
void Xuat()  
{  
    cout << "Ngày tháng năm vừa nhập là: " << iNgay << "/" << iThang <<  
    "/" << iNam << endl;  
}
```

Bảng 3. Nội dung của phương thức Xuat()

1.1.3 - Phương thức NgayThangNamTiepTheo():

- Input (đầu vào): Ngày tháng năm được nhập từ bàn phím/ người dùng.
- Output (đầu ra): Ngày tháng năm được cộng thêm một ngày.
- Hướng giải quyết: Tháng 2 (28/29 ngày) => Tháng có 30 ngày => Tháng có 31 ngày => cuối năm.

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

- Giải quyết tháng 2 đầu tiên vì đặc điểm có năm nhuận:

- Nếu iNam nhập vào chia hết cho 4 (là năm nhuận) => nếu iNgày bằng 29 thì iNgày sẽ bằng 1 và iThang cộng thêm 1, nếu iNgày bé hơn 29 thì iNgày cộng thêm 1.
- Nếu iNam không phải là năm nhuận => nếu iNgày bằng 28 thì iNgày sẽ bằng 1 và iThang cộng thêm 1, nếu iNgày bé hơn 28 thì iNgày cộng thêm 1.

```
18 void NgayThangNamTiepTheo()  
19 {  
20     if (iThang == 2)  
21     {  
22         if (iNam % 4 == 0)  
23         {  
24             if (iNgày == 29)  
25             {  
26                 iNgày = 1;  
27                 iThang++;  
28             }  
29             else if (iNgày < 29)  
30             {  
31                 iNgày++;  
32             }  
33         }  
34     else  
35     {  
36         if (iNgày == 28)  
37         {  
38             iNgày = 1;  
39             iThang++;  
40         }  
41         else if (iNgày < 28)  
42         {  
43             iNgày++;  
44         }  
45     }  
46 }
```

Hình 1.4. Nội dung của hướng giải quyết tháng 2 trong hàm NgayThangNamTiepTheo()

```
void NgayThangNamTiepTheo()  
{  
    if (iThang == 2)  
    {  
        if (iNam % 4 == 0)  
        {  
            if (iNgày == 29)  
            {
```

```
        iNgay = 1;
        iThang++;
    }
    else if (iNgay < 29)
    {
        iNgay++;
    }
}
else
{
    if (iNgay == 28)
    {
        iNgay = 1;
        iThang++;
    }
    else if (iNgay < 28)
    {
        iNgay++;
    }
}
}
```

Bảng 4. Nội dung của hướng giải quyết tháng 2 trong hàm NgayThangNamTiepTheo()

- Giải quyết tháng có 30 ngày:

- Nếu iNgay bằng 30 thì iNgay sẽ bằng 1 và iThang cộng thêm 1, nếu iNgay bé hơn 30 thì iNgay cộng thêm 1.

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
47     else
48     {
49         if (iThang == 4 || iThang == 6 || iThang == 9 || iThang == 11)
50         {
51             if (iNgay == 30)
52             {
53                 iNgay = 1;
54                 iThang++;
55             }
56             else if (iNgay < 30)
57             {
58                 iNgay++;
59             }
60         }
```

Hình 1.5. Nội dung của hướng giải quyết tháng có 30 ngày trong hàm *NgayThangNamTiepTheo()*

```
else
{
    if (iThang == 4 || iThang == 6 || iThang == 9 || iThang == 11)
    {
        if (iNgay == 30)
        {
            iNgay = 1;
            iThang++;
        }
        else if (iNgay < 30)
        {
            iNgay++;
        }
    }
}
```

Bảng 5. Nội dung của hướng giải quyết tháng có 30 ngày trong hàm *NgayThangNamTiepTheo()*

- Giải quyết tháng có 31 ngày:

- Nếu iNgay bằng 31, iThang không phải 12, thì iNgay sẽ bằng 1 và iThang cộng thêm 1, nếu iNgay bé hơn 31 thì iNgay cộng thêm 1.
- Nếu iThang bằng 12, iNgay bé hơn 31 thì iNgay cộng thêm 1, nếu không iNgay, iThang sẽ bằng 1 và iNam cộng thêm 1.

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
61         else
62         {
63             if (iThang == 1 || iThang == 3 || iThang == 5 || iThang ==
64                 7 || iThang == 8 || iThang == 10)
65             {
66                 if (iNgay == 31)
67                 {
68                     iNgay = 1;
69                     iThang++;
70                 }
71                 else if (iNgay < 31)
72                 {
73                     iNgay++;
74                 }
75             }
76             else if (iThang == 12)
77             {
78                 if (iNgay < 31)
79                 {
80                     iNgay++;
81                 }
82                 else
83                 {
84                     iNgay = 1;
85                     iThang = 1;
86                     iNam++;
87                 }
88             }
89         }
90         cout << "Ngày tháng năm tiếp theo là: " << iNgay << '/' << iThang <<
91             '/' << iNam;
92     };
```

Hình 1.6. Nội dung của hướng giải quyết tháng có 31 ngày trong hàm *NgayThangNamTiepTheo()*

```

        else
        {
            if (iThang == 1 || iThang == 3 || iThang == 5 ||
iThang == 7 || iThang == 8 || iThang == 10)
            {
                if (iNgay == 31)
                {
                    iNgay = 1;
                    iThang++;
                }
                else if (iNgay < 31)
                {
                    iNgay++;
                }
            }
            else if (iThang == 12)
            {
                if (iNgay < 31)
                {
                    iNgay++;
                }
                else
                {
                    iNgay = 1;
                    iThang = 1;
                    iNam++;
                }
            }
        }
    }
    cout << "Ngày tháng năm tiếp theo là: " << iNgay << '/' <<
iThang << '/' << iNam;
};
};

```

Bảng 6. Bảng 1.5. Nội dung của hướng giải quyết tháng có 31 ngày trong hàm
NgàyThangNamTiepTheo()

Input: 29 2 2024

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Output:

Ngày tháng năm vừa nhập là: 29/2/2024

Ngày tháng năm tiếp theo là: 1/3/2024

2. Bài tập 2: Xây dựng lớp phân số bao gồm:

- Thuộc tính: iTu, iMau
- Phương thức: Nhap(), Xuat(), RutGon(), Tong(), Hieu(), Tich(), Thuong(), SoSanh()

Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính, phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức trong hàm main().

Class diagram của lớp PhanSo.

Classname
iTu: int iMau: int
+ Nhap(): void + Xuat(): void + RutGon(): void + Tong(): void + Hieu(): void + Tich(): void + Thuong(): void + SoSanh(): void

Hình 2. Class diagram của lớp PhanSo

Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo.

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
1  √ #include <iostream>
2    #include <cstdlib>
3    using namespace std;
4
5  √ class PhanSo
6  {
7  private:
8      int iTu, iMau;
9
10 public:
11 >     void Nhap() ...
15 >     void Xuat() ...
19 >     void RutGon(int &a, int &b) ...
31 >     void Tong(PhanSo b) ...
55 >     void Hieu(PhanSo b) ...
79 >     void Tich(PhanSo b) ...
103 >    void Thuong(PhanSo b) ...
127 >    void Sosanh(PhanSo b) ...
144 };
```

Hình 2.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo

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

class PhanSo
{
private:
    int iTu, iMau;

public:
    void Nhap()...
    void Xuat()...
    void RutGon(int &a, int &b)
    void Tong(PhanSo b)...
    void Hieu(PhanSo b)...
    void Tich(PhanSo b)...
    void Thuong(PhanSo b)...
    void Sosanh(PhanSo b)...
};
```

Bảng 7. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

2.1 Phương thức của class PhanSo:

2.1.1 - Phương thức Nhap():

- Nội dung: Nhập vào từ bàn phím các giá trị iTu, iMau.

```
11      void Nhap()  
12      {  
13          cin >> iTu >> iMau;  
14      }
```

Hình 2.2. Nội dung của phương thức Nhap() của lớp PhanSo

```
void Nhap()  
{  
    cin >> iTu >> iMau;  
}
```

Bảng 8. Nội dung của phương thức Nhap() của lớp PhanSo

2.1.2 - Phương thức Xuat():

- Nội dung: Xuất ra phân số vừa nhập vào bàn phím

```
15      void Xuat()  
16      {  
17          cout << "Phan so da nhap la " << iTu << '/' << iMau << '\n';  
18      }
```

Hình 2.3. Nội dung của phương thức Xuat() của lớp PhanSo

```
void Xuat()  
{  
    cout << "Phan so da nhap la " << iTu << '/' << iMau << '\n';  
}
```

Bảng 9. Nội dung của phương thức Xuat() của lớp PhanSo

2.1.3 - Phương thức RutGon():

- Input (đầu vào): Tử và mẫu sau khi tính toán tổng, hiệu, tích, thương.
- Output (đầu ra): Tử và mẫu sau khi được rút gọn
- Hướng giải quyết: truyền địa chỉ của Tử và Mẫu vào hàm rút gọn (sẽ thay đổi giá trị thực thay vì tạo bản sao). Sử dụng Euclidean algorithm để tìm ước chung

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

lớn nhất => sau đó chia tử vào mẫu cho ước chung lớn nhất ta được phân số rút gọn

```
19 void RutGon(int &a, int &b)
20 {
21     int x = a, y = b;
22     while (y != 0)
23     {
24         int temp = y;
25         y = x % y;
26         x = temp;
27     }
28     a /= x;
29     b /= x;
30 }
```

Hình 2.4. Nội dung của hàm RutGon() của lớp PhanSo()

```
void RutGon(int &a, int &b)
{
    int x = a, y = b;
    while (y != 0)
    {
        int temp = y;
        y = x % y;
        x = temp;
    }
    a /= x;
    b /= x;
}
```

Bảng 10. Nội dung của hàm RutGon() của lớp PhanSo()

Input: 2 4
Output: 1 2

2.1.4 – Phương thức Tong():

- Input (đầu vào): Phân số nhập vào từ bàn phím
- Output (đầu ra): Phân số sau khi tính toán và rút gọn

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức tổng, **kq** sẽ nhận kết quả của tổng đối tượng **this** và **b**. Nếu kết quả của Tử = 0 thì kết quả phép tổng = 0, kết quả Tử = Mẫu thì kết quả của tổng cho ra = 1, **kq** Tử chia hết cho **kq** Mẫu => sử dụng kết quả phép chia /. Trường hợp khác, rút gọn kết quả Tử và Mẫu sau đó in ra phân số.

```
31     void Tong(PhanSo b)
32     {
33         cout << "Tong cua 2 phan so la: ";
34         PhanSo kq;
35         kq.iTu = iTu * b.iMau + iMau * b.iTu;
36         kq.iMau = iMau * b.iMau;
37         if (kq.iTu == 0)
38         {
39             cout << 0 << endl;
40         }
41         else if (kq.iTu == kq.iMau)
42         {
43             cout << 1 << endl;
44         }
45         else if (kq.iTu % kq.iMau == 0)
46         {
47             cout << kq.iTu / kq.iMau << endl;
48         }
49         else
50         {
51             RutGon(kq.iTu, kq.iMau);
52             cout << kq.iTu << '/' << kq.iMau << endl;
53         }
54     }
```

Hình 2.5. Nội dung của phương thức Tong() của lớp PhanSo

```

void Tong(PhanSo b)
{
    cout << "Tong cua 2 phan so la: ";
    PhanSo kq;
    kq.iTu = iTu * b.iMau + iMau * b.iTu;
    kq.iMau = iMau * b.iMau;
    if (kq.iTu == 0)
    {
        cout << 0 << endl;
    }
    else if (kq.iTu == kq.iMau)
    {
        cout << 1 << endl;
    }
    else if (kq.iTu % kq.iMau == 0)
    {
        cout << kq.iTu / kq.iMau << endl;
    }
    else
    {
        RutGon(kq.iTu, kq.iMau);
        cout << kq.iTu << '/' << kq.iMau << endl;
    }
}

```

Bảng 11. Nội dung của phương thức Tong() của lớp PhanSo

Input: 1 2 1 2
Output: 1

2.1.5 – Phương thức Hieu():

- Input (đầu vào): Phân số nhập vào từ bàn phím
- Output (đầu ra): Phân số sau khi tính toán và rút gọn
- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức tổng, **kq** sẽ nhận kết quả của hiệu đối tượng **this** và **b**. Nếu kết quả của Tử = 0 thì kết quả phép hiệu = 0, kết quả Tử = Mẫu thì kết quả của hiệu cho ra = 1, **kq**

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Tử chia hết cho **kq** Mẫu \Rightarrow sử dụng kết quả phép chia /. Trường hợp khác, rút gọn kết quả Tử và Mẫu sau đó in ra phân số.

```
55 void Hieu(PhanSo b)
56 {
57     cout << "Hieu cua 2 phan so la: ";
58     PhanSo kq;
59     kq.iTu = abs(iTu * b.iMau - iMau * b.iTu);
60     kq.iMau = iMau * b.iMau;
61     if (kq.iTu == 0)
62     {
63         cout << 0 << endl;
64     }
65     else if (kq.iTu == kq.iMau)
66     {
67         cout << 1 << endl;
68     }
69     else if (kq.iTu % kq.iMau == 0)
70     {
71         cout << kq.iTu / kq.iMau << endl;
72     }
73     else
74     {
75         RutGon(kq.iTu, kq.iMau);
76         cout << kq.iTu << '/' << kq.iMau << endl;
77     }
78 }
```

Hình 2.6. Nội dung của phương thức Hieu() của lớp PhanSo

```
void Hieu(PhanSo b)
{
    cout << "Hieu cua 2 phan so la: ";
    PhanSo kq;
    kq.iTu = abs(iTu * b.iMau - iMau * b.iTu);
    kq.iMau = iMau * b.iMau;
    if (kq.iTu == 0)
    {
        cout << 0 << endl;
    }
    else if (kq.iTu == kq.iMau)
    {
        cout << 1 << endl;
    }
}
```

```

else if (kq.iTu % kq.iMau == 0)
{
    cout << kq.iTu / kq.iMau << endl;
}
else
{
    RutGon(kq.iTu, kq.iMau);
    cout << kq.iTu << '/' << kq.iMau << endl;
}
}

```

Bảng 12. Nội dung của phương thức Hieu() của lớp PhanSo

Input: 1 2 1 2
Output: 0

2.1.6 – Phương thức Tich():

- Input (đầu vào): Phân số nhập vào từ bàn phím
- Output (đầu ra): Phân số sau khi tính toán và rút gọn
- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức tổng, **kq** sẽ nhận kết quả của tích đối tượng **this** và **b**. Nếu kết quả của Tử = 0 thì kết quả phép tích = 0, kết quả Tử = Mẫu thì kết quả của tích cho ra = 1, **kq** Tử chia hết cho **kq** Mẫu => sử dụng kết quả phép chia /. Trường hợp khác, rút gọn kết quả Tử và Mẫu sau đó in ra phân số.

```
79     void Tich(PhanSo b)
80     {
81         cout << "Tich cua 2 phan so la: ";
82         PhanSo kq;
83         kq.iTu = iTu * b.iTu;
84         kq.iMau = iMau * b.iMau;
85         if (kq.iTu == 0)
86         {
87             cout << 0 << endl;
88         }
89         else if (kq.iTu == kq.iMau)
90         {
91             cout << 1 << endl;
92         }
93         else if (kq.iTu % kq.iMau == 0)
94         {
95             cout << kq.iTu / kq.iMau << endl;
96         }
97         else
98         {
99             RutGon(kq.iTu, kq.iMau);
100            cout << kq.iTu << '/' << kq.iMau << endl;
101        }
102    }
```

Hình 2.7. Nội dung của phương thức Tich() của lớp PhanSo

```
void Tich(PhanSo b)
{
    cout << "Tich cua 2 phan so la: ";
    PhanSo kq;
    kq.iTu = iTu * b.iTu;
    kq.iMau = iMau * b.iMau;
    if (kq.iTu == 0)
    {
        cout << 0 << endl;
    }
    else if (kq.iTu == kq.iMau)
    {
        cout << 1 << endl;
    }
    else if (kq.iTu % kq.iMau == 0)
    {
        cout << kq.iTu / kq.iMau << endl;
    }
}
```

```

    }
    else
    {
        RutGon(kq.iTu, kq.iMau);
        cout << kq.iTu << '/' << kq.iMau << endl;
    }
}

```

Bảng 13. Nội dung của phương thức Tich() của lớp PhanSo

Input: 1 2 1 2
Output: 1 4

2.1.7 – Phương thức Thuong():

- Input (đầu vào): Phân số nhập vào từ bàn phím
- Output (đầu ra): Phân số sau khi tính toán và rút gọn
- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức tổng, **kq** sẽ nhận kết quả của thương đối tượng **this** và **b**. Nếu kết quả của Tử = 0 thì kết quả phép thương = 0, kết quả Tử = Mẫu thì kết quả của thương cho ra = 1, **kq** Tử chia hết cho **kq** Mẫu => sử dụng kết quả phép chia /. Trường hợp khác, rút gọn kết quả Tử và Mẫu sau đó in ra phân số.

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
103     void Thuong(PhanSo b)
104     {
105         cout << "Thuong cua 2 phan so la: ";
106         PhanSo kq;
107         kq.iTu = iTu * b.iMau;
108         kq.iMau = iMau * b.iTu;
109         if (kq.iTu == 0)
110         {
111             cout << 0 << endl;
112         }
113         else if (kq.iTu == kq.iMau)
114         {
115             cout << 1 << endl;
116         }
117         else if (kq.iTu % kq.iMau == 0)
118         {
119             cout << kq.iTu / kq.iMau << endl;
120         }
121         else
122         {
123             RutGon(kq.iTu, kq.iMau);
124             cout << kq.iTu << '/' << kq.iMau << endl;
125         }
126     }
```

Hình 2.8. Nội dung của phương thức Thuong() của lớp PhanSo

```
void Thuong(PhanSo b)
{
    cout << "Thuong cua 2 phan so la: ";
    PhanSo kq;
    kq.iTu = iTu * b.iMau;
    kq.iMau = iMau * b.iTu;
    if (kq.iTu == 0)
    {
        cout << 0 << endl;
    }
    else if (kq.iTu == kq.iMau)
    {
        cout << 1 << endl;
    }
    else if (kq.iTu % kq.iMau == 0)
    {
        cout << kq.iTu / kq.iMau << endl;
    }
}
```



```

    }
    else
    {
        RutGon(kq.iTu, kq.iMau);
        cout << kq.iTu << '/' << kq.iMau << endl;
    }
}

```

Bảng 14. Nội dung của phương thức Thuong() của lớp PhanSo

Input: 1 2 1 2
Output: 1

2.1.8 – Phương thức Sosanh():

- Input (đầu vào): Phân số được nhập từ bàn phím.
- Output (đầu ra): Kết quả so sánh 2 phân số.
- Hướng giải quyết: truyền đối tượng **b**, khởi tạo giá trị thực x, y. Gán kết quả phép chia tử và mẫu của **this** cho x, phép chia tử mẫu của **b** cho y sau đó so sánh 2 giá trị.

```

127     void Sosanh(PhanSo b)
128     {
129         float x = (float)iTu / iMau;
130         float y = (float)b.iTu / b.iMau;
131         if (x > y)
132         {
133             cout << "Phan so thu nhât lon hon phan so thu hai" << endl;
134         }
135         else if (x < y)
136         {
137             cout << "Phan so thu nhât nho hon phan so thu hai" << endl;
138         }
139         else
140         {
141             cout << "2 phan so bang nhau" << endl;
142         }
143     }

```

Hình 2.9. Nội dung của phương thức Sosanh() của lớp PhanSo

<code>void Sosanh(PhanSo b)</code>

```

{
    float x = (float)iTu / iMau;
    float y = (float)b.iTu / b.iMau;
    if (x > y)
    {
        cout << "Phan so thu nhat lon hon phan so thu hai" << endl;
    }
    else if (x < y)
    {
        cout << "Phan so thu nhat nho hon phan so thu hai" << endl;
    }
    else
    {
        cout << "2 phan so bang nhau" << endl;
    }
}

```

Bảng 15. Nội dung của phương thức Sosanh() của lớp PhanSo

Input: 1 5 2 7

Output:

Phan so da nhap la 1/5

Phan so da nhap la 2/7

Tong cua 2 phan so la: 17/35

Hieu cua 2 phan so la: 3/35

Tich cua 2 phan so la: 2/35

Thuong cua 2 phan so la: 7/10

Phan so thu nhat nho hon phan so thu hai
--

3. Bài tập 3: Xây dựng lớp số phức bao gồm:

- Thuộc tính: iThuc, iAo
- Phương thức: Nhap(), Xuat(), Tong(), Hieu(), Tich(), Thuong()

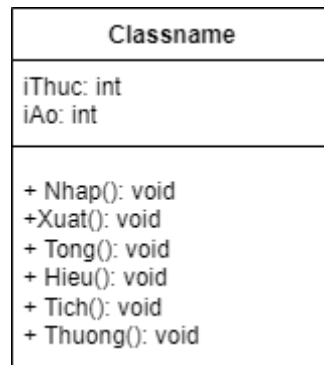
Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính,

phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

trong hàm main()

Class diagram của lớp SoPhuc:



Hình 3. Class diagram của lớp SoPhuc

Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp SoPhuc.

```
1  #include <iostream>
2  #include <iomanip>
3
4  using namespace std;
5
6  class SoPhuc
7  {
8  private:
9      float iThuc, iAo;
10
11 public:
12 > void Nhap() ...
16 > void Xuat() ...
20 > void Tong(SoPhuc b) ...
27 > void Hieu(SoPhuc b) ...
34 > void Tich(SoPhuc b) ...
41 > void Thuong(SoPhuc b) ...
50 };
```

Hình 3.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo.

```

#include <iostream>
#include <iomanip>

using namespace std;

class SoPhuc
{
private:
    float iThuc, iAo;

public:
    void Nhap()...
    void Xuat()...
    void Tong(SoPhuc b)
    void Hieu(SoPhuc b)
    void Tich(SoPhuc b)
    void Thuong(SoPhuc b)
};

```

Bảng 16. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp PhanSo

3.1 Phương thức của class SoPhuc

3.1.1 – Phương thức Nhap():

- Nội dung: Nhập vào bàn phím các giá trị Thực, Ảo

```

12      void Nhap()
13      {
14          cin >> iThuc >> iAo;
15      }

```

Hình 3.2. Nội dung của phương thức Nhap() của lớp SoPhuc

```

void Nhap()
{
    cin >> iThuc >> iAo;
}

```

Bảng 17. Nội dung của phương thức Nhap() của lớp SoPhuc

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

3.1.2 – Phương thức *Xuat()*:

- Nội dung: Xuất ra số phức vừa nhập từ bàn phím.

```
16 void Xuat()  
17 {  
18     cout << "So phuc vua nhap la: " << iThuc << " + " << iAo << 'i' << endl;  
19 }
```

Hình 3.3. Nội dung của phương thức *Xuat()* của lớp *SoPhuc*

```
void Xuat()  
{  
    cout << "So phuc vua nhap la: " << iThuc << " + " << iAo << 'i' <<  
endl;  
}
```

Bảng 18. Nội dung của phương thức *Xuat()* của lớp *SoPhuc*

3.1.3 – Phương thức *Tong()*:

- Input (đầu vào): Số phức nhập vào từ bàn phím.
- Output (đầu ra): Kết quả số phức sau khi tính tổng.
- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức tổng, **kq** sẽ nhận kết quả của tổng đối tượng **this** và **b**.

```
20 void Tong(SoPhuc b)  
21 {  
22     SoPhuc kq;  
23     kq.iThuc = iThuc + b.iThuc;  
24     kq.iAo = iAo + b.iAo;  
25     cout << "Tong cua 2 so phuc la: " << kq.iThuc << " + " << kq.iAo << 'i' << endl;  
26 }
```

Hình 3.4. Nội dung của phương thức *Tong()* trong lớp *SoPhuc*

```
void Tong(SoPhuc b)  
{  
    SoPhuc kq;  
    kq.iThuc = iThuc + b.iThuc;  
    kq.iAo = iAo + b.iAo;  
    cout << "Tong cua 2 so phuc la: " << kq.iThuc << " + " << kq.iAo <<  
'i' << endl;  
}
```

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Bảng 19. Nội dung của phương thức Tong() trong lớp SoPhuc

Input: 2 3 3 2
Output: 5 + 5i

3.1.4 – Phương thức Hieu():

- Input (đầu vào): Số phức nhập vào từ bàn phím.
- Output (đầu ra): Kết quả số phức sau khi tính hiệu.
- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức hiệu, **kq** sẽ nhận kết quả của hiệu đối tượng **this** và **b**.

```
27 void Hieu(SoPhuc b)
28 {
29     SoPhuc kq;
30     kq.iThuc = iThuc - b.iThuc;
31     kq.iAo = iAo - b.iAo;
32     cout << "Hieu cua 2 so phuc la: " << kq.iThuc << " + " << kq.iAo << 'i' << endl;
33 }
```

Hình 3.5. Nội dung của phương thức Hieu() trong lớp SoPhuc

```
void Hieu(SoPhuc b)
{
    SoPhuc kq;
    kq.iThuc = iThuc - b.iThuc;
    kq.iAo = iAo - b.iAo;
    cout << "Hieu cua 2 so phuc la: " << kq.iThuc << " + " << kq.iAo <<
    'i' << endl;
}
```

Bảng 20. Nội dung của phương thức Hieu() trong lớp SoPhuc

Input: 2 3 3 2
Output: -1 + 1i

3.1.5 – Phương thức Tich():

- Input (đầu vào): Số phức nhập vào từ bàn phím.
- Output (đầu ra): Kết quả số phức sau khi tính tích.

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức tích, **kq** sẽ nhận kết quả của tích đối tượng **this** và **b**.

*Lưu ý: 2 phần ảo nhân nhau sẽ nhân thêm -1.

```
34     void Tich(SoPhuc b)
35     {
36         SoPhuc kq;
37         kq.iThuc = iThuc * b.iThuc - iAo * b.iAo;
38         kq.iAo = iThuc * b.iAo + iAo * b.iThuc;
39         cout << "Tich cua 2 so phuc la: " << kq.iThuc << " + " << kq.iAo << 'i' << endl;
40     }
```

Hình 3.6. Nội dung của phương thức Tich() trong lớp SoPhuc

```
void Tich(SoPhuc b)
{
    SoPhuc kq;
    kq.iThuc = iThuc * b.iThuc - iAo * b.iAo;
    kq.iAo = iThuc * b.iAo + iAo * b.iThuc;
    cout << "Tich cua 2 so phuc la: " << kq.iThuc << " + " << kq.iAo <<
'i' << endl;
}
```

Bảng 21. Nội dung của phương thức Tich() trong lớp SoPhuc

Input: 2 3 3 2
Output: 0 + 13i

3.1.6 – Phương thức Thuong():

- Input (đầu vào): Số phức nhập vào từ bàn phím.
- Output (đầu ra): Kết quả số phức sau khi tính thương.
- Hướng giải quyết: truyền đối tượng **b** và tạo đối tượng **kq** trong phương thức thương, **kq** sẽ nhận kết quả của thương đối tượng **this** và **b**.

*Lưu ý: công thức phép chia số phức.

$$\frac{z_1}{z_2} = \frac{z_1 \cdot \overline{z_2}}{|z_2|^2} = \frac{ac + bd}{c^2 + d^2} + \frac{bc - ad}{c^2 + d^2} \cdot i \quad (\text{với } z_2 \neq 0)$$

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
41 void Thuong(SoPhuc b)
42 {
43     int Tu;
44     int Mau;
45     SoPhuc kq;
46     kq.iThuc = (iThuc * b.iThuc + iAo * b.iAo) / (b.iThuc * b.iThuc + b.iAo * b.iAo);
47     kq.iAo = (iAo * b.iThuc - iThuc * b.iAo) / (b.iThuc * b.iThuc + b.iAo * b.iAo);
48     cout << "Thuong cua 2 so phuc la: " << fixed << setprecision(2) << kq.iThuc << " + " << kq.iAo << 'i' << endl;
49 }
50 };
```

Hình 3.7. Nội dung của phương thức Thuong() trong lớp SoPhuc

```
void Thuong(SoPhuc b)
{
    int Tu;
    int Mau;
    SoPhuc kq;
    kq.iThuc = (iThuc * b.iThuc + iAo * b.iAo) / (b.iThuc * b.iThuc +
b.iAo * b.iAo);
    kq.iAo = (iAo * b.iThuc - iThuc * b.iAo) / (b.iThuc * b.iThuc +
b.iAo * b.iAo);
    cout << "Thuong cua 2 so phuc la: " << fixed << setprecision(2) <<
kq.iThuc << " + " << kq.iAo << 'i' << endl;
}
```

Bảng 22. Nội dung của phương thức Thuong() trong lớp SoPhuc

Input: 2 3 3 2

Output:

Tong cua 2 so phuc la: 5 + 5i

Hieu cua 2 so phuc la: -1 + 1i

Tich cua 2 so phuc la: 0 + 13i

Thuong cua 2 so phuc la: 0.92 + 0.38i

4. Bài tập 4: Xây dựng lớp giờ phút giây bao gồm:

- Thuộc tính: iGio, iPhut, iGiay
- Phương thức: Nhap(), Xuat(), TinhCongThemMotGiay()

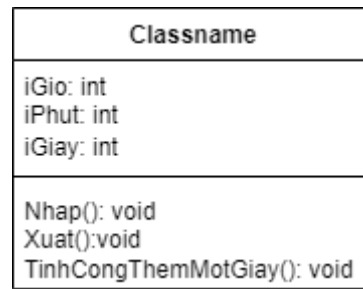
Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính,

phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

trong hàm main()

Class diagram của GioPhutGiay:



Hình 4. Class diagram của lớp GioPhutGiay

Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp GioPhutGiay.

```
1  #include <iostream>
2
3  using namespace std;
4
5  class GioPhutGiay
6  {
7  private:
8      int iGio, iPhut, iGiay;
9
10 public:
11 >     void Nhap() ...
15 >     void Xuat() ...
19 >     void TinhCongThemMotGiay() ...
66 };
```

Hình 4.1. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp GioPhutGiay

```
#include <iostream>

using namespace std;

class GioPhutGiay
{
private:
    int iGio, iPhut, iGiay;
```

```

public:
    void Nhap()...
    void Xuat()...
    void TinhCongThemMotGiay()...
};

```

Bảng 23. Thực hiện xây dựng lớp, khai báo các thuộc tính, phương thức của lớp GioPhutGiay

4.1 Phương thức của class GioPhutGiay

4.1.1 - Phương thức Nhap():

- Nội dung: Nhập vào bàn phím các giá trị iGio, iPhut, iGiay

```

11  void Nhap()
12  {
13      cin >> iGio >> iPhut >> iGiay;
14  }

```

Hình 4.2. Nội dung của phương thức Nhap() của lớp GioPhutGiay

```

void Nhap()
{
    cin >> iGio >> iPhut >> iGiay;
}

```

Bảng 24. Nội dung của phương thức Nhap() của lớp GioPhutGiay

4.1.2 – Phương thức Xuat():

- Nội dung: Xuất ra thời gian sau khi cộng thêm 1 giây

```

15  void Xuat()
16  {
17      cout << "Thoi gian sau khi cong them 1
18      giay la: " << iGio << " gio " << iPhut
        << " phut " << iGiay << " giay\n";
    }

```

Hình 4.3. Nội dung của phương thức Xuat() của lớp GioPhutGiay

```

void Xuat()
{

```

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
cout << "Thời gian sau khi cộng thêm 1 giây là: " << iGio << " gio "
<< iPhut << " phut " << iGiay << " giây\n";
}
```

Bảng 25. Nội dung của phương thức *Xuat()* của lớp *GioPhutGiay*

4.1.3 – Phương thức *TinhCongThemMotGiay()*:

- Input (đầu vào): Giờ phút giây được nhập từ bàn phím/người dùng
- Output (đầu ra): Giờ phút giây sau khi cộng thêm 1 giây
- Hướng giải quyết:

+ TH1: Giây thứ 59, nếu iPhut bé hơn 59 thì iPhut cộng thêm 1, iGiay = 0. Nếu iPhut = 59 mà iGio bé hơn 23 thì iGio cộng thêm 1, iPhut và iGiay = 0. Nếu iPhut = 59 mà iGio = 23 thì iGio = 0, iPhut và iGiay = 0.

+ TH2: Giây bé hơn 59 thì iGiay cộng thêm 1

=> *Xuat()*, nếu iGio,iPhut,iGiay = 0 in ra thêm “Good Morning!”

```
1 void TinhCongThemMotGiay()  
2 {  
3     if (iGiay == 59)  
4     {  
5         iGiay = 0;  
6         if (iPhut < 59)  
7         {  
8             iPhut++;  
9         }  
10        else if (iPhut == 59)  
11        {  
12            iPhut = 0;  
13            if (iGio < 23)  
14            {  
15                iGio++;  
16            }  
17            else if (iGio == 23)  
18            {  
19                iGio = 0;  
20            }  
21            else  
22            {  
23                cout << "Thoi gian ban nhap khong hop le, vui long nhap lai";  
24                exit(0);  
25            }  
26        }  
27        else  
28        {  
29            cout << "Thoi gian ban nhap khong hop le, vui long nhap lai";  
30            exit(0);  
31        }  
32    }  
33    else if (iGiay < 59)  
34    {  
35        iGiay++;  
36    }  
37    else  
38    {  
39        cout << "Thoi gian ban nhap khong hop le, vui long nhap lai";  
40        exit(0);  
41    }  
42    Xuat();  
43    if (iGio == 0 && iPhut == 0 && iGiay == 0)  
44    {  
45        cout << "Good Morning!";  
46    }  
47 }
```

Hình 4.4. Nội dung của phương thức *TinhCongThemMotGiay()* của lớp *GioPhutGiay*

```
void TinhCongThemMotGiay()  
{  
    if (iGiay == 59)  
    {  
        iGiay = 0;  
        if (iPhut < 59)
```

```
{
    iPhut++;
}
else if (iPhut == 59)
{
    iPhut = 0;
    if (iGio < 23)
    {
        iGio++;
    }
    else if (iGio == 23)
    {
        iGio = 0;
    }
    else
    {
        cout << "Thoi gian ban nhap khong hop le, vui long nhap
lai";

        exit(0);
    }
}
else
{
    cout << "Thoi gian ban nhap khong hop le, vui long nhap
lai";

    exit(0);
}
}
else if (iGiay < 59)
{
    iGiay++;
}
else
{
    cout << "Thoi gian ban nhap khong hop le, vui long nhap lai";
    exit(0);
}
}
Xuat();
if (iGio == 0 && iPhut == 0 && iGiay == 0)
```

IT002 – LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

```
{  
    cout << "Good Morning!";  
}  
}
```

Bảng 26. Nội dung của phương thức `TinhCongThemMotGiay()` của lớp `GioPhutGiay`

Input: 23 59 59

Output:

Thời gian sau khi cộng thêm 1 giây là: 0 giờ 0 phút 0 giây

Good Morning!

Drive của file code:

https://drive.google.com/drive/folders/1a8oB6dtpvxbP7JVYV4vAftYSpo_yHfA?usp=sharing