

ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



MÔN : LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG
OBJECT-ORIENTED PROGRAMMING

BÀI TẬP THỰC HÀNH 1

Học viên:
Nguyễn Thiện Nhân –
23521083

Giảng viên:
Nguyễn Ngọc Quý

Mục lục

1	Bài 1	3
2	Bài 2	6
3	Bài 3	9
4	Bài 4	13
5	Bài 5	16
6	Bài 6.1	21
7	Bài 6.2	24
8	Bài 7	32
9	Bài 8	43
10	Phụ lục	51
11	Lời cảm ơn	51

Danh sách hình vẽ

1	<i>Phân số được rút gọn</i>	4
2	<i>Phân số âm</i>	5
3	<i>Phân số chứa 0</i>	5
4	<i>Hai phân số dương</i>	8
5	<i>Hai phân số âm</i>	8
6	<i>Hai phân số bằng nhau</i>	8
7	<i>Hai phân số dương</i>	11
8	<i>Có phân số âm</i>	12
9	<i>Hai phân số bằng nhau</i>	12
10	<i>Trường hợp bình thường</i>	15
11	<i>Trường hợp bình thường</i>	15
12	<i>Trường hợp bình thường</i>	15
13	<i>Ngày bình thường</i>	19
14	<i>Ngày đầu năm</i>	20
15	<i>Ngày cuối năm</i>	20
16	<i>Trường hợp ngẫu nhiên</i>	23
17	<i>Xuất hiện ở mọi vị trí</i>	23
18	<i>Trường hợp bình thường</i>	30
19	<i>Trường hợp bình thường</i>	31
20	<i>Kiểm thử các trường hợp không hợp lệ và chức năng tìm kiếm đầu tiên</i>	40
21	<i>kiểm thử chức năng tìm kiếm thứ 2</i>	41
22	<i>kiểm thử chức năng tìm kiếm thứ 3</i>	42
23	<i>Kiểm thử các trường hợp không hợp lệ và chức năng tìm kiếm đầu tiên</i>	49
24	<i>kiểm thử với nhiều khách hàng</i>	50

1 Bài 1

Đề bài: Viết chương trình nhập vào một phân số, rút gọn phân số và xuất kết quả

- Input: Phân số được nhập từ bàn phím
- Output: Phân số được rút gọn
- Xây dựng lớp Fraction
 - Thuộc tính : Numer, Deno
 - Phương thức : Input(), Shorten()
 - * Input() : Nhập dữ liệu
 - * Shorten() : Rút gọn phân số bằng việc tìm GCD và chia
- Solution : Sử dụng lớp Fraction để lưu dữ liệu và dùng Shorten để xử lý
- Code:

```
#include <bits/stdc++.h>

using namespace std;

class Fraction {
private:
    // Numer : tu so
    // Deno : mau so
    int Numer, Deno;
public:
    Fraction(int number, int deno) {
        Numer = number;
        Deno = deno;
    }

    void Input() {
```

```

        cout << "Nhap tu so va mau so : ";
        cin >> Numer >> Deno;
    }

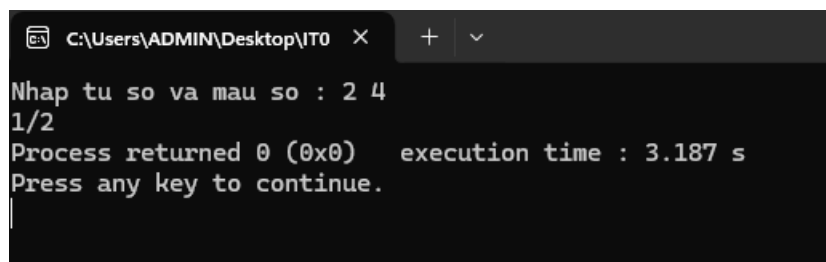
    void Shorten() {
        int tmp = __gcd(Numer, Deno);
        Numer /= tmp;
        Deno /= tmp;
    }

    void Prin() {
        cout << Numer << "/" << Deno;
    }
};

int main() {
    Fraction a(1,1);
    a.Input();
    a.Shorten();
    a.Prin();
    return 0;
}

```

- Các testcase :

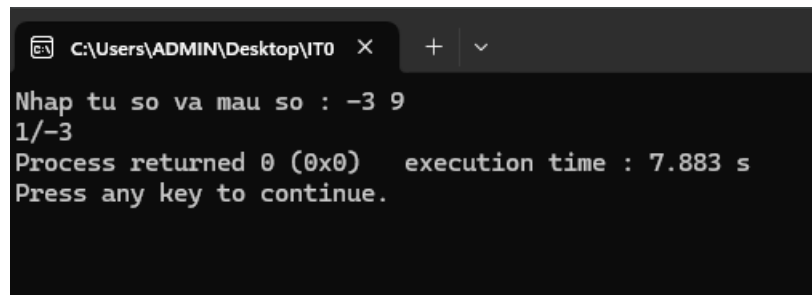


```

C:\Users\ADMIN\Desktop\IT0 >
Nhap tu so va mau so : 2 4
1/2
Process returned 0 (0x0)   execution time : 3.187 s
Press any key to continue.

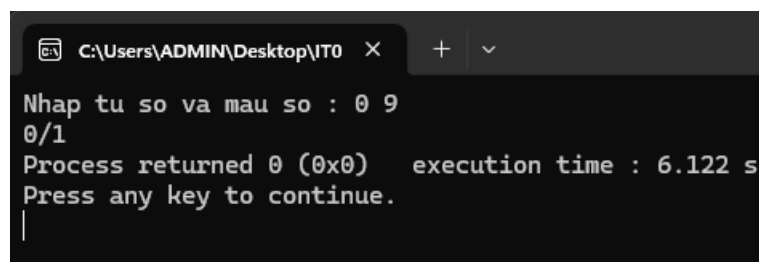
```

Hình 1: Phân số được rút gọn



```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap tu so va mau so : -3 9
1/-3
Process returned 0 (0x0) execution time : 7.883 s
Press any key to continue.
```

Hình 2: Phân số âm



```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap tu so va mau so : 0 9
0/1
Process returned 0 (0x0) execution time : 6.122 s
Press any key to continue.
|
```

Hình 3: Phân số chứa 0

2 Bài 2

Đề bài: Viết chương trình nhập vào hai phân số, tìm phân số lớn nhất và xuất kết quả.

- Input: Tử số và mẫu số của 2 phân số
- Output: Phân số lớn hơn (Nếu hai phân số bằng nhau hoặc có một tử số bằng 0 thì xuất ra thông báo)
- Xây dựng lớp Fraction
 - Thuộc tính : Numer, Deno
 - Phương thức : Input(), Shorten(), <
 - * Input() : Nhập dữ liệu
 - * Shorten() : Rút gọn phân số bằng việc tìm GCD và chia
 - * < : Định nghĩa lại phép so sánh
- Solution : Sử dụng lớp Fraction để lưu dữ liệu và so sánh
- Code :

```
#include <bits/stdc++.h>

using namespace std;

class Fraction {
private:
    // Numer : tu so
    // Deno : mau so
    int Numer, Deno;
public:
    Fraction(int numer, int deno) {
        Numer = numer;
        Deno = deno;
    }
};
```

```

    }

    void Input() {
        cout << "Nhap tu so va mau so : ";
        cin >> Numer >> Deno;
    }

    void Shorten() {
        int tmp = __gcd(Numer, Deno);
        Numer /= tmp;
        Deno /= tmp;
    }

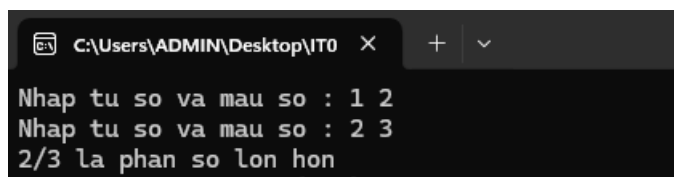
    // Dinh nghia lai phep so sanh
    //  $a/b < c/d \Leftrightarrow a * d < b * c$ 
    bool operator < (const Fraction& other) const {
        return (Numer * other.Deno < other.Numer * Deno);
    }

    void Prin() {
        cout << Numer << "/" << Deno;
    }
};

int main() {
    Fraction a(1,1), b(1, 1);
    a.Input(); b.Input();
    if (a < b) {
        b.Prin();
        cout << " la phan so lon hon";
    } else
        if (b < a) {
            a.Prin();
            cout << " la phan so lon hon";
        } else
            cout << "Hai phan so bang nhau";
    return 0;
}

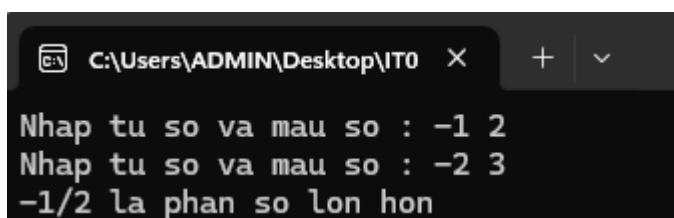
```

- Testcase :



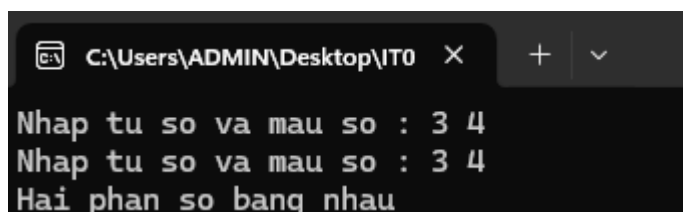
```
C:\Users\ADMIN\Desktop\IT0 > Nhap tu so va mau so : 1 2
C:\Users\ADMIN\Desktop\IT0 > Nhap tu so va mau so : 2 3
C:\Users\ADMIN\Desktop\IT0 > 2/3 la phan so lon hon
```

Hình 4: Hai phân số dương



```
C:\Users\ADMIN\Desktop\IT0 > Nhap tu so va mau so : -1 2
C:\Users\ADMIN\Desktop\IT0 > Nhap tu so va mau so : -2 3
C:\Users\ADMIN\Desktop\IT0 > -1/2 la phan so lon hon
```

Hình 5: Hai phân số âm



```
C:\Users\ADMIN\Desktop\IT0 > Nhap tu so va mau so : 3 4
C:\Users\ADMIN\Desktop\IT0 > Nhap tu so va mau so : 3 4
C:\Users\ADMIN\Desktop\IT0 > Hai phan so bang nhau
```

Hình 6: Hai phân số bằng nhau

3 Bài 3

Đề bài: Viết chương trình nhập vào hai phân số. Tính tổng, hiệu, tích, thương giữa chúng và xuất kết quả.

- Input: Hai phân số được nhập từ bàn phím
- Output: Tổng hiệu tích thương của hai phân số
- Xây dựng lớp Fraction
 - Thuộc tính Numer, Deno
 - Phương thức : Input(), Prin(), <, +, -, *, /
 - * Input() : Nhập dữ liệu
 - * Prin() : Xuất dữ liệu
 - * <, +, -, *, / : định nghĩa lại các phép toán và so sánh
- Solution : Dùng Input() để nhập dữ liệu, dựa trên những phép toán đã được định nghĩa ta tạo các biến lưu giá trị và xuất ra.
- Code:

```
#include <bits/stdc++.h>

using namespace std;

class Fraction {
private:
    // Numer : tu so
    // Deno : mau so
    int Numer, Deno;
public:
    Fraction(int numer, int deno) {
        Numer = numer;
        Deno = deno;
    }
}
```

```

void Input() {
    cout << "Nhap tu so va mau so : ";
    cin >> Numer >> Deno;
}

void Shorten() {
    int tmp = __gcd(abs(Numer), abs(Deno));
    Numer /= tmp;
    Deno /= tmp;
}

// Dinh nghia lai phep so sanh
// a/b < c/d <=> a * d < b * c
bool operator < (const Fraction& other) const {
    return (Numer * other.Deno < other.Numer * Deno);
}

Fraction operator + (const Fraction& other) const {
    return Fraction(Numer * other.Deno + other.Numer *
        Deno, Deno * other.Deno);
}

Fraction operator - (const Fraction& other) const {
    return Fraction(Numer * other.Deno - other.Numer *
        Deno, Deno * other.Deno);
}

Fraction operator * (const Fraction& other) const {
    return Fraction(Numer * other.Numer, Deno *
        other.Deno);
}

Fraction operator / (const Fraction& other) const {
    return Fraction(Numer * other.Deno, Deno *
        other.Numer);
}

void Prin() {
    cout << Numer << "/" << Deno << "\n";
}

```

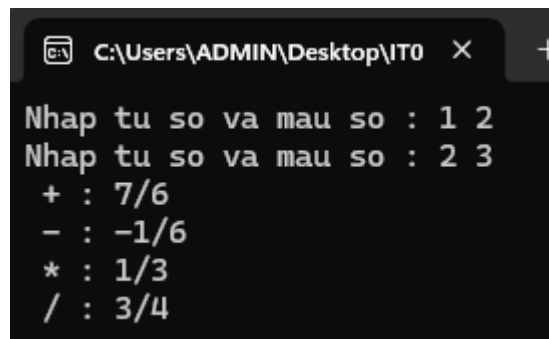
```

    }
};

int main() {
    Fraction a(1,1), b(1, 1);
    a.Input(); b.Input();
    Fraction Plus = a + b;
    Fraction Minus = a - b;
    Fraction Times = a * b;
    Fraction Div = a / b;
    cout << " + : "; Plus.Shorten(); Plus.Prin();
    cout << " - : "; Minus.Shorten(); Minus.Prin();
    cout << " * : "; Times.Shorten(); Times.Prin();
    cout << " / : "; Div.Shorten(); Div.Prin();
    return 0;
}

```

- Testcase :



```

C:\Users\ADMIN\Desktop\IT0
Nhập tử số và mẫu số : 1 2
Nhập tử số và mẫu số : 2 3
+ : 7/6
- : -1/6
* : 1/3
/ : 3/4

```

Hình 7: Hai phân số dương

```
C:\Users\ADMIN\Desktop\IT0 X +
Nhap tu so va mau so : 2 3
Nhap tu so va mau so : -1 4
+ : 5/12
- : 11/12
* : -1/6
/ : 8/-3
```

Hình 8: Có phân số âm

```
C:\Users\ADMIN\Desktop\IT0 X +
Nhap tu so va mau so : 1 1
Nhap tu so va mau so : 1 1
+ : 2/1
- : 0/1
* : 1/1
/ : 1/1
```

Hình 9: Hai phân số bằng nhau

4 Bài 4

Đề bài: Lập chương trình tính $\sin(x)$ với độ chính xác 0.00001 theo công thức

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5} + \dots + \frac{(-1)^n x^{2n+1}}{(2n+1)!}$$

- Input : Số thực x
- Output: Giá trị của $\sin(x)$ với độ chính xác 0.00001
- Xây dựng các hàm
 - Factorial(int n) : hàm tính $n!$
 - Hàm POW(long long a, long long b) : hàm tính a^b
 - Hàm Sin(int x) : áp dụng công thức đề bài và sử dụng hàm Factorial và POW để tính $\sin(x)$
- Solution : Xây dựng hàm Sin(int x) dựa trên hàm Factorial(int n) và POW(long long a, long long b) để giải quyết
- Code:

```
#include <bits/stdc++.h>

using namespace std;

// Input : n; Output : n!
long long Factorial(int n) {
    long long ans = 1;
    for (int i = 1; i <= n; i++)
        ans = ans * i;
    return ans;
}
```

```

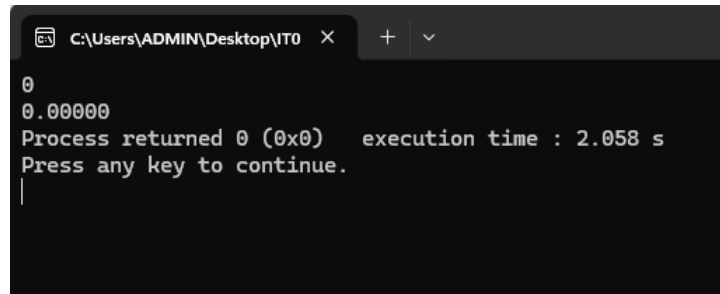
// Input (a, b); Output : a ^ b
long long POW(long long a, long long b) {
    if (b == 0)
        return 1;
    if (b == 1)
        return a;
    long long tmp = POW(a, b / 2);
    if (b % 2)
        return tmp * tmp % a;
    else
        return tmp * tmp;
}

// Input : x; Output : sin(x)
double Sin(int x) {
    double ans = 0, n = 0;
    while (true) {
        double cur = POW(-1, n) * POW(x, 2 * n + 1) /
            Factorial(2 * n + 1);
        ans = ans + cur;
        if (cur < 0.00001)
            break;
    }
    return ans;
}

int main() {
    int n;
    cin >> n;
    cout << setprecision(5) << fixed;
    cout << sin(n);
    return 0;
}

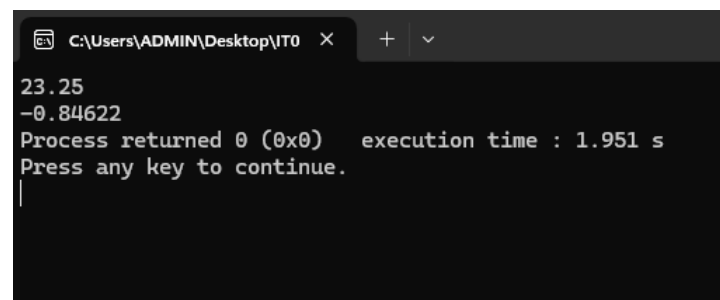
```

- Testcase :



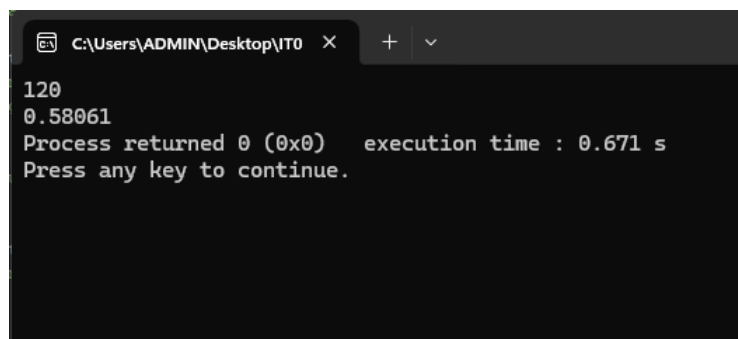
```
C:\Users\ADMIN\Desktop\IT0 X + v
0
0.00000
Process returned 0 (0x0) execution time : 2.058 s
Press any key to continue.
|
```

Hình 10: Trường hợp bình thường



```
C:\Users\ADMIN\Desktop\IT0 X + v
23.25
-0.84622
Process returned 0 (0x0) execution time : 1.951 s
Press any key to continue.
|
```

Hình 11: Trường hợp bình thường



```
C:\Users\ADMIN\Desktop\IT0 X + v
120
0.58061
Process returned 0 (0x0) execution time : 0.671 s
Press any key to continue.
|
```

Hình 12: Trường hợp bình thường

5 Bài 5

Đề bài: Viết chương trình nhập vào một ngày.

- Tìm ngày kế tiếp và xuất kết quả.
- Tìm ngày trước ngày này và xuất kết quả.
- Tính xem ngày đó là ngày thứ bao nhiêu trong năm và xuất kết quả.

- Input: Ngày, tháng, năm của một ngày
- Output: Ngày hôm trước, ngày hôm sau và thứ tự trong năm của ngày đó
- Xây dựng lớp ToDay
 - Thuộc tính : Day, Month, Year
 - Phương thức : Today(int day, int month, int year), InputData(), NextDay(Today other), LastDay(Today other), CountDay(), Prin()
 - * Today(int day, int month, int year) : Khởi tạo, gán dữ liệu
 - * InputData : Nhập dữ liệu từ bàn phím
 - * NextDay(Today other) : Nhận vào một ngày và trả về ngày tiếp theo
 - * LastDay(Today other) : Nhận vào một ngày và trả về ngày hôm trước
 - * CountDay() : Trả về giá trị là số thứ tự của ngày đó trong năm
 - * Prin() : In ra ngày
- Solution :
 - Lập ra một mảng lưu số ngày của mỗi tháng

- Tăng số ngày lên một thì xét có vượt quá số ngày trong tháng không nếu có thì xem tháng và năm thay đổi như thế nào
- Giảm số ngày xuống thì kiểm tra xem nếu giảm về 0 thì thay đổi luôn tháng và năm
- Tính tổng số ngày của các tháng trước đó và cộng với tháng hiện tại

- Code:

```
// de bai khong de cap nen xin phep khong lam nam nhuan
// de nghi khong nhap so nam la 1
#include <bits/stdc++.h>

using namespace std;

const int Numday[12] = {31, 28, 31, 30, 31, 30, 31, 31,
                        30, 31, 30, 31};

class ToDay {
private:
    int Day, Month, Year;
public:
    ToDay(int day, int month, int year) {
        Day = day;
        Month = month;
        Year = year;
    }

    void InputData() {
        cout << "Nhap vao ngay : ";
        cin >> Day;
        cout << "Nhap vao thang : ";
        cin >> Month;
        cout << "Nhap vao nam : ";
        cin >> Year;
    }

    static ToDay NextDay(ToDay other) {
        other.Day++;
```

```

        if (other.Day > Numday[other.Month - 1]) {
            other.Day = 1;
            other.Month++;
            if (other.Month == 13) {
                other.Month = 1;
                other.Year++;
            }
        }
        return other;
    }

    static ToDay LastDay(ToDay other) {
        other.Day--;
        if (other.Day <= 0) {
            other.Month--;
            if (other.Month <= 0) {
                other.Month = 12;
                other.Year--;
            }
            other.Day = Numday[other.Month - 1];
        }
        return other;
    }

    int CountDay() {
        int ans = 0;
        for (int i = 0; i < Month - 1; i++) {
            ans = ans + Numday[i];
        }
        ans = ans + Day;
        return ans;
    }

    void Prin() {
        cout << Day << "/" << Month << "/" << Year << "\n";
    }
};

int main() {
    ToDay CurDay(0, 0, 0);

```

```

CurDay.InputData();

ToDay a = ToDay::NextDay(CurDay);
a.Prin();

ToDay b = ToDay::LastDay(CurDay);
b.Prin();

int Cnt = CurDay.CountDay();
cout << Cnt;

return 0;
}

```

- Testcase :

```

C:\Users\ADMIN\Desktop\IT0 >
Nhap vao ngay : 2
Nhap vao thang : 1
Nhap vao nam : 2005
3/1/2005
1/1/2005
2
Process returned 0 (0x0)   execution time : 8.637 s
Press any key to continue.

```

Hình 13: Ngày bình thường

```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap vao ngay : 1
Nhap vao thang : 1
Nhap vao nam : 2024
2/1/2024
31/12/2023
1
Process returned 0 (0x0)   execution time : 4.651 s
Press any key to continue.
|
```

Hình 14: Ngày đầu năm

```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap vao ngay : 31
Nhap vao thang : 12
Nhap vao nam : 2024
1/1/2025
30/12/2024
365
Process returned 0 (0x0)   execution time : 2.594 s
Press any key to continue.
```

Hình 15: Ngày cuối năm

6 Bài 6.1

Đề bài : Viết chương trình nhập vào hai mảng A, B
Ghi ra số lần xuất hiện của mảng A trong B

- Input : Dòng đầu tiên chứa số phần tử của mảng A và mảng B.
 - Dòng thứ hai chứa các phần tử của mảng A.
 - Dòng thứ ba chứa các phần tử của mảng B.
- Output : Số lần mảng A xuất hiện trong mảng B.
 - Danh sách các chỉ số bắt đầu của những lần xuất hiện
- Xây dựng lớp Array
 - Thuộc tính : n, Arr
 - Phương thức : Array(int n, vector <int>&arr), Find_In(Array&tmp)
 - * Array(int n, vector <int> & arr) : Khởi tạo, gán
 - * Find_In(Array& tmp) : Truyền vào mảng tmp, trả về vector các lần xuất hiện của tmp trong mảng chính
- Solution : Để tìm số lần số hiện của một mảng trên một mảng khác, ta bắt đầu từ mỗi vị trí của mảng cha và chạy xem có giống với mảng con hay không, nếu một phần tử khác lập tức break ra. Nếu giống hết thì ghi nhận kết quả
- Code :

```
#include <bits/stdc++.h>

using namespace std;

// Voi mot thuat toan don gian de hieu la O(n ^ 2)
// Nen chi lam viec voi mang < 10000

class Array {
private:
    int N;
    vector <int> Arr;
```

```

public:
    Array(int n, vector<int>&arr) {
        N = n;
        Arr = arr;
    }

    // Input : mang tmp
    // Output : vector vi tri xuat hien cua tmp trong mang
    cha
    vector<int> Find_In(Array& tmp) {
        vector<int> ans;
        for (int i = 0; i + tmp.N - 1 < N; i++) {
            bool check = true;
            for (int j = 0; j < tmp.N; j++)
                if (tmp.Arr[j] != Arr[i + j]) {
                    check = false;
                    break;
                }
            if (check)
                ans.push_back(i);
        }
        return ans;
    }
};

int main() {
    int n1, n2;
    vector<int> a, b;
    cin >> n1 >> n2;
    for (int i = 0; i < n1; i++) {
        int cur;
        cin >> cur;
        a.push_back(cur);
    }
    for (int i = 0; i < n2; i++) {
        int cur;
        cin >> cur;
        b.push_back(cur);
    }
}

```

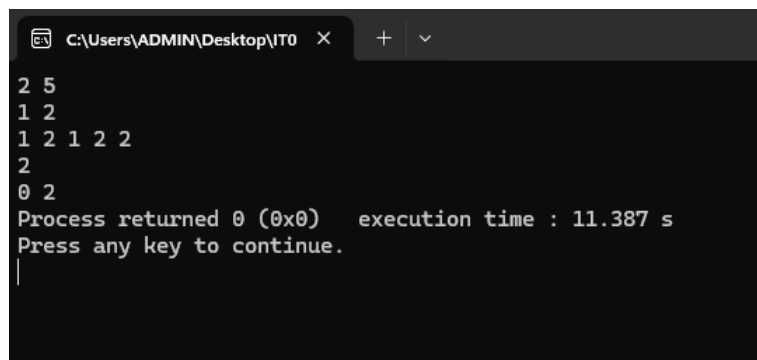
```

    Array A(n1, a); Array B(n2, b);

    vector <int> ans = B.Find_In(A);
    cout << ans.size() << "\n";
    for (auto cur : ans)
        cout << cur << " ";
    return 0;
}

```

- Testcase :

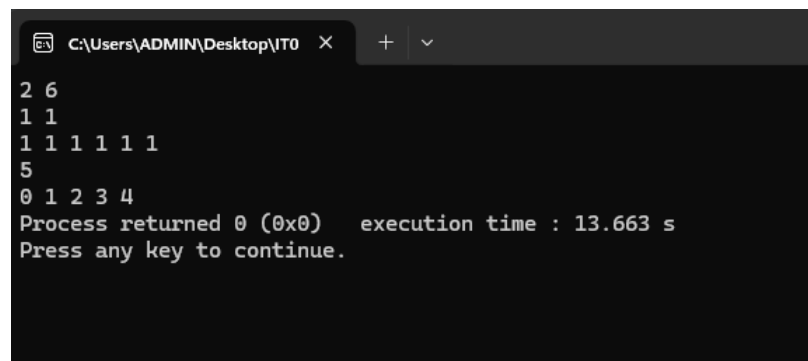


```

C:\Users\ADMIN\Desktop\IT0 >
2 5
1 2
1 2 1 2 2
2
0 2
Process returned 0 (0x0)   execution time : 11.387 s
Press any key to continue.

```

Hình 16: Trường hợp ngẫu nhiên



```

C:\Users\ADMIN\Desktop\IT0 >
2 6
1 1
1 1 1 1 1 1
5
0 1 2 3 4
Process returned 0 (0x0)   execution time : 13.663 s
Press any key to continue.

```

Hình 17: Xuất hiện ở mọi vị trí

7 Bài 6.2

Đề bài: Viết chương trình nhập vào 3 học sinh

- Điểm trung bình và phân loại học sinh.
- Học sinh có điểm trung bình cao nhất.
- Kết quả tìm kiếm theo tên.
- Danh sách học sinh có điểm toán thấp nhất

- Input : 3 học sinh gồm điểm toán, văn, ngoại ngữ, học sinh muốn tìm kiếm
- Output : Điểm trung bình, phân loại của học sinh, học sinh có điểm trung bình cao nhất, những học sinh có điểm toán thấp nhất và kết quả tìm kiếm theo tên của học sinh
- Xây dựng lớp Student
 - Thuộc tính : Name, Math, Lir, Eng, Avg;
 - Phương thức : NameCheck(string&Tmp), CheckPoint(double val), GetPoint(double&val), GetAvg(), Type(), BetterAvg(Student a, Student b), <, SameMath(sutdent a), TheBest(), Ok(), Prin(), Input()
 - * NameCheck(string&Tmp) : Truyền vào một tên trả về tên đó có hợp lệ hay không
 - * CheckPoint(double val) : Truyền vào một số trả về đó có phải điểm hợp lệ hay không
 - * GetPoint(double&val) : Hàm nhập điểm cho đến khi hợp lệ
 - * GetAvg() : Hàm tính điểm trung bình
 - * Type() : Hàm phân loại học sinh theo điểm trung bình và in ra kết quả phân loại

- * BetterAvg(Student a, Student b) : Truyền vào hai học sinh trả về học sinh có điểm trung bình cao hơn
- * < định nghĩa lại phép so sánh dựa trên điểm toán
- * SameMath(Student a) : truyền vào một học sinh và trả về liệu học sinh hiện tại có cùng điểm toán với học sinh truyền vào hay không
- * TheBest() : In ra đây là học sinh có điểm trung bình cao nhất
- * Ok(string tmp) : truyền vào một chuỗi và trả về liệu chuỗi đó có nằm trong tên hay không
- * Prin() : Hàm in ra tên, điểm trung bình, điểm toán của học sinh
- * Input(): Hàm nhập tên, điểm, nhắc nhở nếu không hợp lệ và nhập cho đến khi đúng

– Solution :

- * Kiểm tra tên có hợp lệ hay không bằng cách xem có xuất hiện số hay kí tự đặc biệt gì không
- * Kiểm tra điểm có nằm trong khoảng từ 0 đến 10 hay không
- * Xây dựng hàm kiểm tra học sinh có điểm trung bình cao và duyệt qua tất cả các học sinh
- * Định nghĩa lại cách so sánh các học sinh dựa trên điểm toán, sắp xếp lại các học sinh và ghi ra những học sinh có điểm toán thấp nhất
- * Viết hoa hết các kí tự ở tên và tìm kiếm để kiểm tra xem tên cần tìm có nằm trong tên học sinh hay không, duyệt qua và ghi ra kết quả tìm kiếm

• Code :

```
#include <bits/stdc++.h>
```

```

using namespace std;

class Student {
private:
    string Name;
    double Math, Lir, Eng, Avg;
public:

    // Kiem tra xem co chua cac ki tu khong phai chu cai
    hay khong
    bool NameCheck(string&Tmp) {
        if (Tmp == "")
            return false;
        for (char cur : Tmp) {
            if (cur == ' ')
                continue;
            if (!('a' <= cur && cur <= 'z') && !('A' <= cur
                && cur <= 'Z'))
                return false;
        }
        return true;
    }

    // Kiem tra diem hop le
    bool CheckPoint(double val) {
        if (val < 0 || 10 < val)
            return false;
        return true;
    }

    void GetPoint(double&val) {
        while (true) {
            cin >> val;
            if (CheckPoint(val))
                break;
            else
                cout << "Khong hop le yeu cau nhap lai \n";
        }
    }
}

```

```

void GetAvg() {
    Avg = (Math * 2 + Lir + Eng) / 4;
}

// ham phan loai hoc sinh
void Type() {
    cout << setprecision(2) << fixed;
    cout << "Hoc sinh : " << Name << " diem trung binh
        " << Avg << " xep loai : ";
    if (Avg < 5)
        cout << "Yeu";
    else
        if (Avg < 6.5)
            cout << "Trung Binh";
        else
            if (Avg < 8)
                cout << "Kha";
            else
                if (Avg < 9)
                    cout << "Gioi";
                else
                    cout << "Xuat sac";
    cout << "\n";
}

// ham tra ve hoc sinh co diem trung binh cao hon
static Student BetterAvg(Student a, Student b) {
    if (a.Avg >= b.Avg)
        return a;
    else
        return b;
}

// ham so sanh dua tren diem toan
bool operator < (const Student& a) const {
    return (Math < a.Math);
}

bool SameMath(Student a) {
    return (Math == a.Math);
}

```

```

}

// xuất ra học sinh có điểm trung bình cao nhất
void TheBest() {
    cout << "Học sinh có điểm cao nhất là : " << Name
        << "\n";
}

bool Ok(string tmp) {
    string cur = Name;
    for (char&c : cur)
        c = toupper(c);
    for (char&c : tmp)
        c = toupper(c);
    if (cur.find(tmp) != -1)
        return true;
    else
        return false;
}

void Prin() {
    cout << Name << " điểm trung bình : " << Avg << "
        << "\n";
}

void Input() {
    while (true) {
        cout << "Nhập tên học sinh : ";
        getline(cin, Name);
        if (NameCheck(Name))
            break;
        else
            cout << "Không hợp lệ yêu cầu nhập lại \n";
    }
    cout << "Nhập điểm toán : ";
    GetPoint(Math);
    cout << "Nhập điểm văn : ";
    GetPoint(Lir);
    cout << "Nhập điểm ngoại ngữ : ";
}

```

```

        GetPoint(Eng);
        cin.ignore();
        GetAvg();
    }
};

int main() {
    int n = 3;
    Student a[n];
    for (int i = 0; i < n; i++)
        a[i].Input();

    for (int i = 0; i < n; i++)
        a[i].Type();

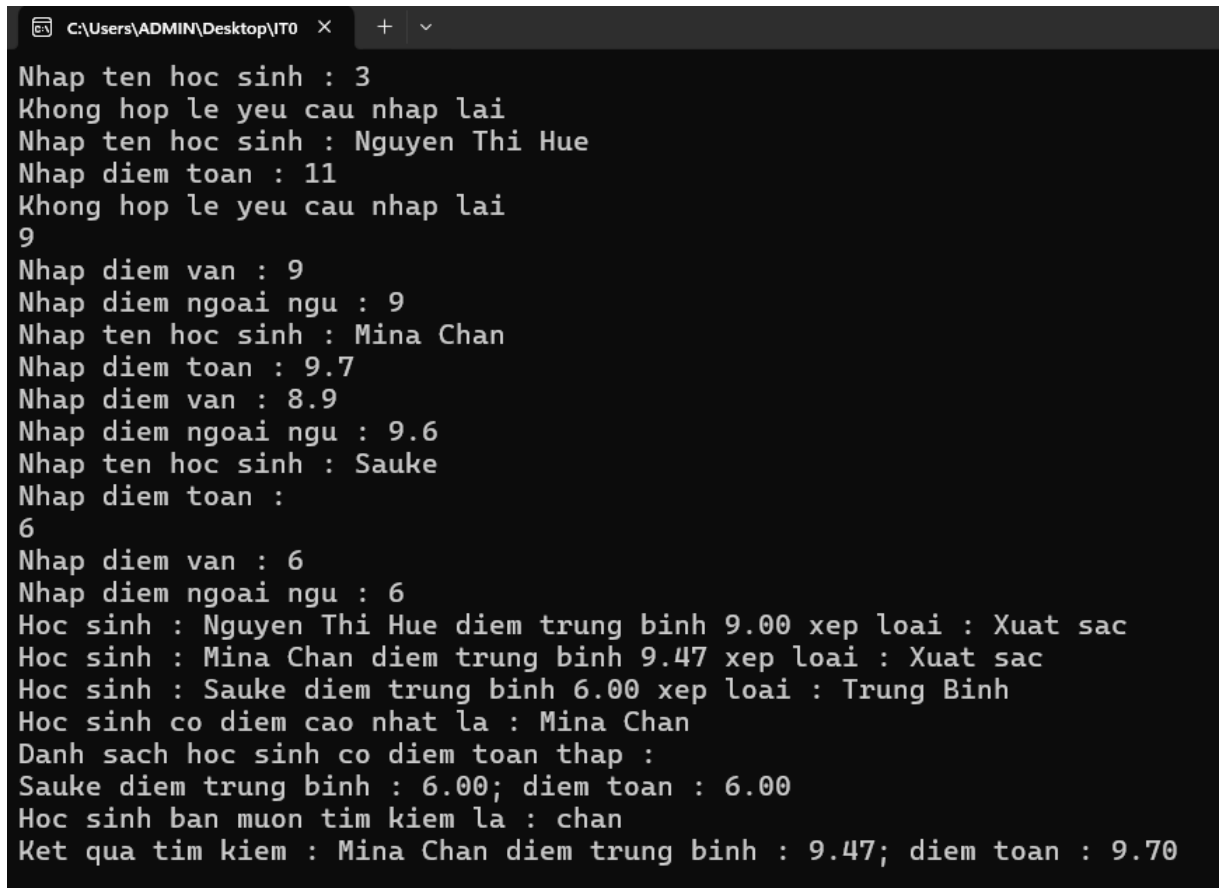
    Student Best = a[0];
    for (int i = 0; i < n; i++)
        Best = Student::BetterAvg(Best, a[i]);
    Best.TheBest();

    cout << "Danh sach hoc sinh co diem toan thap :" <<
        "\n";
    sort(a + 0, a + 0 + n);
    for (int i = 0; i < n; i++)
        if (a[i].SameMath(a[0]))
            a[i].Prin();

    string s;
    cout << "Hoc sinh ban muon tim kiem la : ";
    getline(cin, s);
    cout << "Ket qua tim kiem : ";
    for (int i = 0; i < n; i++)
        if (a[i].Ok(s))
            a[i].Prin();
    return 0;
}

```

- Testcase :



```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap ten hoc sinh : 3
Khong hop le yeu cau nhap lai
Nhap ten hoc sinh : Nguyen Thi Hue
Nhap diem toan : 11
Khong hop le yeu cau nhap lai
9
Nhap diem van : 9
Nhap diem ngoai ngu : 9
Nhap ten hoc sinh : Mina Chan
Nhap diem toan : 9.7
Nhap diem van : 8.9
Nhap diem ngoai ngu : 9.6
Nhap ten hoc sinh : Sauke
Nhap diem toan :
6
Nhap diem van : 6
Nhap diem ngoai ngu : 6
Hoc sinh : Nguyen Thi Hue diem trung binh 9.00 xep loai : Xuat sac
Hoc sinh : Mina Chan diem trung binh 9.47 xep loai : Xuat sac
Hoc sinh : Sauke diem trung binh 6.00 xep loai : Trung Binh
Hoc sinh co diem cao nhat la : Mina Chan
Danh sach hoc sinh co diem toan thap :
Sauke diem trung binh : 6.00; diem toan : 6.00
Hoc sinh ban muon tim kiem la : chan
Ket qua tim kiem : Mina Chan diem trung binh : 9.47; diem toan : 9.70
```

Hình 18: Trường hợp bình thường

```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap ten hoc sinh : Naruto
Nhap diem toan : 5
Nhap diem van : 6.9
Nhap diem ngoai ngu : 9
Nhap ten hoc sinh : Sakura
Nhap diem toan : 9
Nhap diem van : 9
Nhap diem ngoai ngu : 9
Nhap ten hoc sinh : Sasuke
Nhap diem toan : 5
Nhap diem van : 5
Nhap diem ngoai ngu : 5
Hoc sinh : Naruto diem trung binh 6.47 xep loai : Trung Binh
Hoc sinh : Sakura diem trung binh 9.00 xep loai : Xuat sac
Hoc sinh : Sasuke diem trung binh 5.00 xep loai : Trung Binh
Hoc sinh co diem cao nhat la : Sakura
Danh sach hoc sinh co diem toan thap :
Naruto diem trung binh : 6.47; diem toan : 5.00
Sasuke diem trung binh : 5.00; diem toan : 5.00
Hoc sinh ban muon tim kiem la : Sa
Ket qua tim kiem : Sasuke diem trung binh : 5.00; diem toan : 5.00
Sakura diem trung binh : 9.00; diem toan : 9.00

Process returned 0 (0x0)   execution time : 36.444 s
Press any key to continue.
|
```

Hình 19: Trường hợp bình thường

8 Bài 7

Đề bài:

Hãy xây dựng một chương trình quản lý thông tin chuyến bay. Mỗi chuyến bay bao gồm các thông tin sau:

- Mã chuyến bay: Chuỗi ký tự dài tối đa 5 ký tự, không được chứa khoảng trắng hay ký tự đặc biệt.
- Ngày bay: Dữ liệu ngày khởi hành, yêu cầu kiểm tra tính hợp lệ của ngày nhập (ngày phải tồn tại và hợp lệ theo lịch).
- Giờ bay: Dữ liệu giờ khởi hành, được nhập theo định dạng 24 giờ (từ 00:00 đến 23:59).
- Nơi đi: Chuỗi ký tự dài tối đa 20 ký tự. Phải là tên địa danh hợp lệ, không được chứa số hoặc ký tự đặc biệt.
- Nơi đến: Chuỗi ký tự dài tối đa 20 ký tự. Tương tự như nơi đi, yêu cầu kiểm tra định dạng.

Yêu cầu:

1. Viết các hàm nhập và xuất dữ liệu cho một chuyến bay.
 - Kiểm tra và báo lỗi nếu người dùng nhập sai các định dạng yêu cầu.
2. Viết thêm các chức năng sau:
 - Tìm kiếm chuyến bay theo mã chuyến bay, nơi đi hoặc nơi đến.
 - Sắp xếp danh sách chuyến bay theo thứ tự ngày và giờ khởi hành (từ sớm đến muộn).
3. Hiển thị danh sách tất cả các chuyến bay khởi hành từ một nơi cụ thể trong một ngày được chỉ định.
4. Đếm số lượng chuyến bay từ một nơi đi đến một nơi đến nhất định.

- Input : Số lượng chuyến bay, thông tin của chuyến bay (mã, ngày giờ, nơi cất cánh và hạ cánh của máy bay), thông tin muốn tìm kiếm (có thể là mã, nơi cất cánh và ngày bay, nơi cất cánh và nơi hạ cách)
- Output : Các chuyến bay đã được sắp xếp và các chuyến bay đáp ứng như cầu tìm kiếm
- Xây dựng lớp Flight :
 - Thuộc tính : Id, Day, Month, Year, Sec, Hour, From, To;

- Phương thức : IdCheck(), TimeCheck(), DayCheck(), PlaceCheck(string Place), Input(), Prin(), Find1(string tmp), Find2(string XuatPhat, int Ngay, int Thang, int Nam), Find3(string XuatPhat, string KetThuc)
 - * IdCheck() : Trả về kết quả xem mã chuyến bay có hợp lệ hay không
 - * TimeCheck() : Trả về giờ đó có hợp lệ hay không
 - * DayCheck() : Trả về ngày đó có hợp lệ hay không
 - * PlaceCheck(string Place) : Trả về tên địa danh có hợp lệ hay không
 - * Input() : Hàm nhập, trong đó gọi các hàm kiểm tra xem thông tin có hợp lệ hay không nếu không thông báo để nhập lại
 - * Prin() : In ra chuyến bay
 - * Find1(string tmp) : truyền vào một mã trả về mã chuyến bay có giống với mã truyền vào hay không
 - * Find2(string XuatPhat, int Ngay, int Thang, int Nam) : Truyền vào điểm cất cánh và ngày tháng năm trả về xem chuyến bay có khớp hay không
 - * Find3(string XuatPhat, string KetThuc) : Truyền vào điểm cất cánh và điểm hạ cánh trả về nó có hợp với chuyến bay hiện tại hay không
 - * < : định nghĩa lại phép so sánh theo thứ tự ưu tiên năm, tháng, ngày, giờ, phút

• Solution :

- Để in ra danh sách các chuyến bay thì ta sắp xếp dựa trên phép so sánh đã định nghĩa lại

- Khi có một thông tin cần tra cứu ta đơn giản duyệt qua tất cả các chuyến bay trong danh sách và chọn phương thức kiểm tra phù hợp như đã viết ở trên

- Code :

```
#include <bits/stdc++.h>

using namespace std;

const int Numday[12] = {31, 28, 31, 30, 31, 30, 31, 31,
    30, 31, 30, 31};

class Flight {
private:
    string Id;
    int Day, Month, Year;
    int Sec, Hour;
    string From;
    string To;
public:
    bool IdCheck();
    bool TimeCheck();
    bool DayCheck();
    bool PlaceCheck(string Place);
    void Input();
    void Prin();
    bool Find1(string tmp);
    bool Find2(string XuatPhat, int Ngay, int Thang, int
        Nam);
    bool Find3(string XuatPhat, string KetThuc);

    bool operator < (const Flight& a) const {
        if (Year != a.Year)
            return Year < a.Year;
        if (Month != a.Month)
            return Month < a.Month;
        if (Day != a.Day)
            return Day < a.Day;
        if (Hour != a.Hour)
            return Hour < a.Hour;
```

```

        if (Sec != a.Sec)
            return Sec < a.Sec;
        return From < a.From;
    }
};

bool Flight :: IdCheck() {
    if (Id.size() > 5)
        return false;
    for (char c : Id) {
        if (!isalnum(c))
            return false;
    }
    return true;
}

bool Flight :: PlaceCheck(string Place) {
    if (Place.size() > 20) {
        return false;
    }
    for (char c : Place) {
        if (!isalnum(c) && c != ' ')
            return false;
    }
    return true;
}

bool Flight :: DayCheck() {
    if (Day <= 0 || Month <= 0 || Year <= 0)
        return false;
    if (Month > 12)
        return false;
    if (Day > Numday[Month - 1])
        return false;
    return true;
}

bool Flight :: TimeCheck() {
    if (Sec < 0 || Hour < 0)
        return false;

```

```

        if (Sec > 59 || Hour > 23)
            return false;
        return true;
    }

    void Flight :: Input() {
        while (true) {
            cout << "Nhap ma chuyen bay : ";
            getline(cin, Id);
            if (IdCheck())
                break;
            else
                cout << "Khong hop le yeu cau nhap lai \n";
        }

        while (true) {
            cout << "Nhap ngay bay theo dinh dang ngay _ thang
                _ nam : ";
            cin >> Day >> Month >> Year;
            if (DayCheck())
                break;
            else
                cout << "Khong hop le yeu cau nhap lai \n";
        }

        while (true) {
            cout << "Nhap gio bay theo dinh dang gio _ phut :
                ";
            cin >> Hour >> Sec;
            if (TimeCheck())
                break;
            else
                cout << "Khong hop le yeu cau nhap lai \n";
        }

        cin.ignore();

        while (true) {
            cout << "Nhap diem cat cach : ";
            getline(cin, From);

```

```

        if (PlaceCheck(From))
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }

    while (true) {
        cout << "Nhap diem ha canh : ";
        getline(cin, To);
        if (PlaceCheck(To))
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }
}

void Flight :: Prin() {
    cout << "Chuyen bay " << Id << " khoi hanh vao luc "
        << Day << "-" << Month << "-" << Year << " " <<
        Hour << ":" << Sec << " di tu "
        << From << " den " << To << "\n";
}

bool Flight :: Find1(string tmp) {
    return (tmp == Id);
}

bool Flight :: Find2(string XuatPhat, int Ngay, int Thang,
    int Nam) {
    return ((XuatPhat == From && Ngay == Day) && (Thang ==
        Month && Year == Nam));
}

bool Flight :: Find3(string XuatPhat, string KetThuc) {
    return (XuatPhat == From && KetThuc == To);
}

int main() {

    int n;

```

```

cout << "Nhap so luong chuyen bay : ";
cin >> n; cin.ignore();
Flight Plane[n];
for (int i = 0; i < n; i++) {
    cout << "\n";
    Plane[i].Input();
}

cout << '\n';
cout << "Cac chuyen bay sau khi sap xep theo gio khoi
    hanh la : \n";
sort(Plane + 0, Plane + 0 + n);
for (int i = 0; i < n; i++) {
    Plane[i].Prin();
}

cout << "\n";
cout << "Chon hinh thuckiem tra : \n";
cout << "1 Nhap vao ma chuyen bay dekiem tra \n";
cout << "2 Nhap vao diem xuat phat va ngay bay \n";
cout << "3 Dem so luong chuyen bay tu diem xuat phat
    va diem ket thuc \n";
int type;
cin >> type;

if (type == 1) {
    cout << "Moi nhap ma chuyen bay : ";
    string st; bool ok = false;
    cin >> st;

    for (Flight cur : Plane) {
        if (cur.Find1(st))
        {
            ok = true;
            cur.Prin();
        }
    }
    if (!ok)
        cout << "Chuyen bay khong ton tai";
}

```

```

if (type == 2) {
    string XuatPhat;
    int Ngay, Thang, Nam;
    cout << "Moi nhap diem xuat phat :";
    cin.ignore();
    getline(cin, XuatPhat);
    cout << "Nhap ngay bay theo dinh dang ngay _ thang
        _ nam : ";
    cin >> Ngay >> Thang >> Nam;

    vector <Flight> ans;
    for (Flight cur : Plane)
        if (cur.Find2(XuatPhat, Ngay, Thang, Nam))
            ans.push_back(cur);
    cout << "Co " << ans.size() << " chuyen bay duoc
        tim thay, sau day la danh sach : \n";
    for (Flight cur : ans)
        cur.Prin();
}

if (type == 3) {
    string XuatPhat, KetThuc;
    cout << "Moi nhap diem xuat phat :";
    cin.ignore();
    getline(cin, XuatPhat);
    cout << "Moi nhap diem den :";
    getline(cin, KetThuc);

    int ans = 0;
    for (Flight cur : Plane)
        if (cur.Find3(XuatPhat, KetThuc))
            ans++;
    cout << "Co " << ans << " chuyen bay khop voi
        thong tin tim kiem : \n";
}
return 0;
}

```

- Testcase :

```

C:\Users\ADMIN\Desktop\IT0 x + v
Nhap so luong chuyen bay : 3

Nhap ma chuyen bay : @Vn12
Khong hop le yeu cau nhap lai
Nhap ma chuyen bay : VN123
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 50 12 2024
Khong hop le yeu cau nhap lai
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 12 12 2024
Nhap gio bay theo dinh dang gio _ phut : 25 11
Khong hop le yeu cau nhap lai
Nhap gio bay theo dinh dang gio _ phut : 23 59
Nhap diem cat cach : Vung Dat mang ten thu do Ha Noi
Khong hop le yeu cau nhap lai
Nhap diem cat cach : Ha Noi
Nhap diem ha canh : @Ho Chi Minh
Khong hop le yeu cau nhap lai
Nhap diem ha canh : Tp Ho Chi Minh

Nhap ma chuyen bay : VN234
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 02 01 2024
Nhap gio bay theo dinh dang gio _ phut : 23 59
Nhap diem cat cach : Ha Noi
Nhap diem ha canh : Hue

Nhap ma chuyen bay : VN456
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 02 01 2024
Nhap gio bay theo dinh dang gio _ phut : 12 12
Nhap diem cat cach : Hue
Nhap diem ha canh : Tp Ho Chi Minh

Cac chuyen bay sau khi sap xep theo gio khoi hanh la :
Chuyen bay VN456 khoi hanh vao luc 2-1-2024 12:12 di tu Hue den Tp Ho Chi Minh
Chuyen bay VN234 khoi hanh vao luc 2-1-2024 23:59 di tu Ha Noi den Hue
Chuyen bay VN123 khoi hanh vao luc 12-12-2024 23:59 di tu Ha Noi den Tp Ho Chi Minh

Chon hinh thuc kiem tra :
1 Nhap vao ma chuyen bay de kiem tra
2 Nhap vao diem xuat phat va ngay gio
3 Dem so luong chuyen bay tu diem xuat phat va diem ket thuc
1
Moi nhap ma chuyen bay : VN123
Chuyen bay VN123 khoi hanh vao luc 12-12-2024 23:59 di tu Ha Noi den Tp Ho Chi Minh

```

Hình 20: Kiểm thử các trường hợp không hợp lệ và chức năng tìm kiếm đầu tiên

```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap so luong chuyen bay : 3

Nhap ma chuyen bay : VN123
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 02 01 2005
Nhap gio bay theo dinh dang gio _ phut : 12 39
Nhap diem cat cach : Quang Tri
Nhap diem ha canh : Ho Chi Minh

Nhap ma chuyen bay : VN234
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 02 01 2005
Nhap gio bay theo dinh dang gio _ phut : 15 30
Nhap diem cat cach : Quang Tri
Nhap diem ha canh : Tokyo

Nhap ma chuyen bay : VN987
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 02 01 2005
Nhap gio bay theo dinh dang gio _ phut : 17 00
Nhap diem cat cach : Ho Chi Minh
Nhap diem ha canh : Tokyo

Cac chuyen bay sau khi sap xep theo gio khoi hanh la :
Chuyen bay VN123 khoi hanh vao luc 2-1-2005 12:39 di tu Quang Tri den Ho Chi Minh
Chuyen bay VN234 khoi hanh vao luc 2-1-2005 15:30 di tu Quang Tri den Tokyo
Chuyen bay VN987 khoi hanh vao luc 2-1-2005 17:0 di tu Ho Chi Minh den Tokyo

Chon hinh thuckiem tra :
1 Nhap vao ma chuyen bay dekiem tra
2 Nhap vao diemxuat phat va ngay bay
3 Dem so luong chuyen bay tu diemxuat phat va diem ket thuc
2
Moi nhap diemxuat phat :Quang Tri
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 02 01 2005
Co 2 chuyen bay duoc tim thay, sau day la danh sach :
Chuyen bay VN123 khoi hanh vao luc 2-1-2005 12:39 di tu Quang Tri den Ho Chi Minh
Chuyen bay VN234 khoi hanh vao luc 2-1-2005 15:30 di tu Quang Tri den Tokyo
```

Hình 21: kiểm thử chức năng tìm kiếm thứ 2

```

Nhap so luong chuyen bay : 3

Nhap ma chuyen bay : VN123
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 2 2 2002
Nhap gio bay theo dinh dang gio _ phut : 2 2
Nhap diem cat cach : Ha Noi
Nhap diem ha canh : Quang Tri

Nhap ma chuyen bay : VN2
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 2 2 2005
Nhap gio bay theo dinh dang gio _ phut : 2 2
Nhap diem cat cach : Ha Noi
Nhap diem ha canh : Hue

Nhap ma chuyen bay : Vn3
Nhap ngay bay theo dinh dang ngay _ thang _ nam : 2 2 2005
Nhap gio bay theo dinh dang gio _ phut : 2 2
Nhap diem cat cach : Ha Noi
Nhap diem ha canh : Quang Tri

Cac chuyen bay sau khi sap xep theo gio khai hanh la :
Chuyen bay VN123 khai hanh vao luc 2-2-2002 2:2 di tu Ha Noi den Quang Tri
Chuyen bay VN2 khai hanh vao luc 2-2-2005 2:2 di tu Ha Noi den Hue
Chuyen bay Vn3 khai hanh vao luc 2-2-2005 2:2 di tu Ha Noi den Quang Tri

Chon hinh thuc kiem tra :
1 Nhap vao ma chuyen bay de kiem tra
2 Nhap vao diem xuat phat va ngay bay
3 Dem so luong chuyen bay tu diem xuat phat va diem ket thuc
3
Moi nhap diem xuat phat :Ha Noi
Moi nhap diem den :Quang Tri
Co 2 chuyen bay khop voi thong tin tim kiem :

```

Hình 22: kiểm thử chức năng tìm kiếm thứ 3

9 Bài 8

Viết chương trình nhập số tiết kiệm (Do quá dài và quá nản nên em đã tin giảm một chút ở bài này)

- Input : Số lượng khách, thông tin khách hàng, thông tin tìm kiếm
- Output : Sắp xếp khách hàng và kết quả tìm kiếm
- Xây dựng lớp Flight :
 - Thuộc tính : Id, Type, Name, CMND, Day, Month, Year, Money
 - Phương thức : IdCheck(), TypeCheck(), NameCheck(), CMND-Check(), DayCheck(), MoneyChec(), Input(), Prin(), Find(string st), <
 - * IdCheck() : Trả về kết quả xem mã số tiết kiệm có hợp lệ không
 - * TypeCheck() : Trả về kết quả xem loại tiết kiệm hợp lệ không
 - * NameCheck() : Trả về kết quả tên có hợp lệ hay không
 - * CMNDCheck() : Trả về kết quả xem trên có hợp lệ hay không
 - * DayCheck() : Trả về kết quả xem ngày có hợp lệ không
 - * MoneyCheck() : Trả về kết quả xem tiền có hợp lệ không
 - * Input() : Nhập thông tin khách hàng
 - * Prin() : In thông tin khách hàng
 - * Find(string st) : Kiểm tra xem st có khớp với Id hoặc CMND hay không

- Solution :
 - Dùng mảng SaveMoney để lưu thông tin khách hàng và sắp xếp
 - Duyệt hết mảng để tra cứu
 - Code :
-

```
#include <bits/stdc++.h>

using namespace std;

const int Numday[12] = {31, 28, 31, 30, 31, 30, 31, 31,
                        30, 31, 30, 31};

class SaveMoney {
private:
    string Id;
    string Type;
    string Name;
    string CMND;
    int Day, Month, Year;
    double Money;
public:
    bool IdCheck();
    bool TypeCheck();
    bool NameCheck();
    bool CMNDCheck();
    bool DayCheck();
    bool MoneyCheck();
    void Input();
    void Prin();
    bool Find(string st);
    bool operator < (const SaveMoney& a) const {
        return Money < a.Money;
    }
};

bool SaveMoney :: IdCheck() {
    if (Id.size() > 5)
        return false;
}
```

```

    for (char c : Id) {
        if (!isalnum(c))
            return false;
    }
    return true;
}

bool SaveMoney :: TypeCheck() {
    return (Type.size() <= 10);
}

bool SaveMoney :: NameCheck() {
    if (Name.size() > 30) {
        return false;
    }
    for (char c : Name) {
        if (!isalnum(c) && c != ' ')
            return false;
    }
    return true;
}

bool SaveMoney :: CMNDCheck() {
    if (CMND.size() != 9 && CMND.size() != 12)
        return false;
    for (char c : CMND) {
        if (!isdigit(c))
            return false;
    }
    return true;
}

bool SaveMoney :: DayCheck() {
    if (Day <= 0 || Month <= 0 || Year <= 0)
        return false;
    if (Month > 12)
        return false;
    if (Day > Numday[Month - 1])
        return false;
    return true;
}

```

```

}

bool SaveMoney :: MoneyCheck() {
    return (Money >= 0);
}

void SaveMoney :: Input() {
    while (true) {
        cout << "Nhap ma so : ";
        getline(cin, Id);
        if (IdCheck())
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }

    while (true) {
        cout << "Nhap loai tiet kiem : ";
        getline(cin, Type);
        if (TypeCheck())
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }

    while (true) {
        cout << "Nhap ten khach hang : ";
        getline(cin, Name);
        if (NameCheck())
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }

    while (true) {
        cout << "Nhap chung minh nhan dan : ";
        getline(cin, CMND);
        if (CMNDCheck())
            break;
    }
}

```

```

        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }

    while (true) {
        cout << "Nhap ngay _ thang _ nam ";
        cin >> Day >> Month >> Year;
        if (DayCheck())
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }

    while (true) {
        cout << "Nhap so tien gui : ";
        cin >> Money;
        if (MoneyCheck())
            break;
        else
            cout << "Khong hop le yeu cau nhap lai \n";
    }
}

void SaveMoney :: Prin() {
    cout << "Khach hang " << Name << ", so chung minh nhan
        dan " << CMND << ", ma so tiet kiem " << Id << ",
        loai tiet kiem " << Type << ", gui vao ngay " << Day
        << "/" << Month << "/" << Year << " voi so tien la : "
        << Money << " d\n";
}

bool SaveMoney :: Find(string st)
{
    return (st == Id || st == CMND);
}

int main() {
    int n;
    cout << "Nhap so luong khach hang : ";
    cin >> n;

```



```

SaveMoney p[n];
for (int i = 0; i < n; i++)
{
    cin.ignore();
    p[i].Input();
}

sort(p + 0, p + 0 + n);
cout << "Danh sach khach hang sau khi da sap xep theo
        so tien : \n";
for (int i = n - 1; i >= 0; i--)
    p[i].Prin();

cout << "Nhap so chung minh nhan dan hoac ma so tiet
        kiem ban muon tim kiem :\n";
string st; cin >> st;
for (auto cur : p)
    if (cur.Find(st))
    {
        cout << "Day la khach hang ban can tim \n";
        cur.Prin();
        return 0;
    }
cout << "Khong tim thay khach hang trung khop";
return 0;
}

```

- Testcase :

```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap so luong khách hàng : 1
Nhap mã số : @123
Không hợp lệ yêu cầu nhập lại
Nhap mã số : 123
Nhap loại tiết kiệm : tiết kiệm nhiều nhất
Không hợp lệ yêu cầu nhập lại
Nhap loại tiết kiệm : tiết kiệm
Nhap tên khách hàng : Nguyễn Thị Đẹp Tuyet Tran Thong Minh Thong Thai Viet Cho Dai
Không hợp lệ yêu cầu nhập lại
Nhap tên khách hàng : Nguyễn Văn A
Nhap chứng minh nhân dân : 123456
Không hợp lệ yêu cầu nhập lại
Nhap chứng minh nhân dân : 123456789
Nhap ngày _ tháng _ năm 23 24 2024
Không hợp lệ yêu cầu nhập lại
Nhap ngày _ tháng _ năm 23 12 2024
Nhap số tiền gửi : -1
Không hợp lệ yêu cầu nhập lại
Nhap số tiền gửi : 1000
Danh sách khách hàng sau khi đã sắp xếp theo số tiền :
Khách hàng Nguyễn Văn A, số chứng minh nhân dân 123456789, mã số tiết kiệm 123, loại tiết kiệm tiết kiệm, gửi
Nhap số chứng minh nhân dân hoặc mã số tiết kiệm bạn muốn tìm kiếm :
123
Đây là khách hàng bạn cần tìm
Khách hàng Nguyễn Văn A, số chứng minh nhân dân 123456789, mã số tiết kiệm 123, loại tiết kiệm tiết kiệm, gửi

Process returned 0 (0x0)   execution time : 72.845 s
Press any key to continue.
```

Hình 23: Kiểm thử các trường hợp không hợp lệ và chức năng tìm kiếm đầu tiên

```
C:\Users\ADMIN\Desktop\IT0 X + v
Nhap so luong khách hàng : 2
Nhap mã số : 123
Nhap loại tiết kiệm : tiết kiệm
Nhap tên khách hàng : Nguyen Van Nhat
Nhap chứng minh nhân dân : 123456789
Nhap ngày _ tháng _ năm 2 1 2005
Nhap số tiền gửi : 10000000000000
Nhap mã số : 345
Nhap loại tiết kiệm : giao
Nhap tên khách hàng : Nguyen Thi Hai
Nhap chứng minh nhân dân : 123456789123
Nhap ngày _ tháng _ năm 2 1 2006
Nhap số tiền gửi : 10000000000
Danh sách khách hàng sau khi đã sắp xếp theo số tiền :
Khách hàng Nguyen Van Nhat, số chứng minh nhân dân 123456789, mã số tiết kiệm 123, loại tiết
: 1e+013 d
Khách hàng Nguyen Thi Hai, số chứng minh nhân dân 123456789123, mã số tiết kiệm 345, loại tiết
1e+009 d
Nhap số chứng minh nhân dân hoặc mã số tiết kiệm bạn muốn tìm kiếm :
123456789123
Đây là khách hàng bạn cần tìm
Khách hàng Nguyen Thi Hai, số chứng minh nhân dân 123456789123, mã số tiết kiệm 345, loại tiết
1e+009 d

Process returned 0 (0x0)   execution time : 59.560 s
Press any key to continue.
|
```

Hình 24: kiểm thử với nhiều khách hàng

10 Phụ lục

Bạn có thể xem kĩ hơn ở : [LAB1_00P](#)

11 Lời cảm ơn

Em xin chân thành cảm ơn sự hướng dẫn và giúp đỡ của thầy trong suốt quá trình thực hành môn lập trình hướng đối tượng. Mong thầy đánh giá và cho điểm cao ạ.