Chủ đề: Vận dụng kiến thức Lập trình cơ bản với C++ để giải quyết các bài toán thực tế

Trong mỗi môn học về lập trình, việc áp dụng để giải bài toán thực tế là yếu tố quan trọng để phát triển kỹ năng lập trình. Việc phát hiện vấn đề và tư duy xử lý vấn đề luôn là yếu tố và chìa khóa quan trọng để đánh giá được năng lực của người học trong quá trình học lập trình

Trong học phần này, cũng như các học phần tiếp theo (Cấu trúc dữ liệu và giải thuật, Lập trình hướng đối tượng), sẽ luôn có các bài toán vận dụng vào giải quyết vấn đề thực tiễn dựa trên các kiến thức đã học của học phần, theo mức độ khó sẽ tăng dần.

1. **Kiến thức tiền quyết**

* Các kiến thức cơ bản về C++: cú pháp, biến, giá trị của biến, toán tử, comment, trình bày code
* Các cấu trúc điều khiển: rẽ nhánh, lặp
* Các cấu trúc dữ liệu: mảng (tĩnh, động), các thuật toán thao tác
* Con trỏ và cấp phát động
* String, các thao tác với string
* Thao tác với file văn bản có dạng .txt với file cpp
* Các kiến thức về lý thuyết số: phân biệt, xác định tính chất, loại số,….

1. **Tư duy giải quyết vấn đề thực tế**

* Để giải quyết các bài toán thực tế ứng dụng các kiến thức lập trình, ta cần suy nghĩ đến các yếu tố sau:

+ **Yếu tố trường hợp:** Các trường hợp trong tình huống, mà khi dính phải các trường hợp đặc biệt đó, chúng ta phải có cách xử lý riêng.

Chúng ta có thể khái quát các trường hợp và phương hướng giải quyết qua sơ đồ thuật toán.

VD: Một con thuyền có khả năng chở tối đa k người, và phải chở cùng 1 lúc n người qua sông. Nếu k < n thì việc này không thực hiện được. Còn nếu

k n thì có thể đưa qua

+ **Vấn đề lưu trữ và sử dụng dữ liệu:** Trong các bài toán, việc lưu trữ và sử dụng dữ liệu vô cùng quan trọng, quyết định tính tối ưu và mức độ khoa học của chương trình.

VD: Nếu bài chỉ có 1,2 biến cơ bản, bạn có thể khai báo biến thông thường. Tuy nhiên nếu lưu trữ nhiều giá trị, việc lưu mỗi giá trị một biến riêng lẻ sẽ khó để truy xuất. Bạn có thể sử dụng các cấu trúc dữ liệu như mảng (tĩnh hoặc động). Trong việc sử dụng mảng tĩnh hay động, mảng động được sử dụng ưu tiên để quản lý bộ nhớ khoa học hơn. Mặt khác, phụ thuộc vào độ lớn của dữ liệu, các kiểu dữ liệu cũng khiến cho chương trình trở nên “gọn” hơn trong bộ nhớ

+ **Vấn đề thuật toán**: Sau khi giải quyết 2 vấn đề trên, ta sẽ suy luận thuật toán để giải quyết vấn đề. Thuật toán có thể do bạn nghĩ ra, hoặc là các thuật toán đã học từ các vấn đề trong quá trình học lý thuyết. Tuy nhiên, thuật toán vẫn phải đảm bảo đúng, khoa học và không có kẽ hở, để chúng có thể xử lý vấn đề một cách trọn vẹn.

+ **Vấn đề tổ chức chương trình:** Việc tổ chức chương trình khoa học và hiệu quả giúp chính bạn quản lý chương trình tốt hơn, và người khác khi muốn đọc code và nghiên cứu code của bạn sẽ dễ quản lý hơn. Đây là kỹ năng cơ bản đối với bất kỳ lập trình viên nào.

Để tổ chức chương trình khoa học, bạn có thể sử dụng các cấu trúc như hàm, class method,…

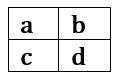
VD: Cùng 1 đoạn chương trình, nhưng thay vì bạn phải viết đi viết lại nhiều lần trong chương trình, bạn có thể viết chúng qua một hàm, và trong hàm main(), bạn sẽ gọi lại hàm đó với đối số truyền vào (nếu có)

Nói chung, để giải quyết tốt các bài toán thực tế trong lập trình, bạn cần giành nhiều thời gian để xem xét và học hỏi, nhất là kỹ năng tự học. Hiện nay có rất nhiều trang tutorial giúp bạn luyện kỹ năng này như Codelearn, Codecademy, Leetcode,….

1. **Ví dụ về các bài toán và hướng phân tích chương trình**

Đây chỉ là bài toán ví dụ để minh họa cách giải quyết vấn đề, trên thực tế có vô số cách giải quyết khác nhau, và vô số bài toán khác nhau.

VD: Cho 4 số nguyên dương a, b, c ,d được viết vào bảng vuông kích thước 2 x 2 theo thứ tự sau:



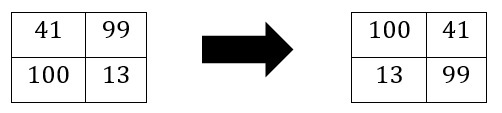
Ta gọi giá trị của bảng trên là



Hãy xác định số lượt quay 90 độ theo chiều kim đồng hồ để được bảng có giá trị lớn nhất. Nếu có nhiều cách quay bảng để có được giá trị lớn nhất thì chỉ ra số lượt quay ít nhất.

Ví dụ:

Với a = 41, b = 99, c = 100, d = 13 kết quả sẽ là rotateTable(a, b, c, d) = 1



1. Phân tích bài toán:

* Bài toán yêu cầu tìm số lượt quay 90 độ theo chiều kim đồng hồ để được bảng có giá trị lớn nhất. Trên thực tế, để xác định xem đâu là số lượt quay tối thiểu, chúng ta phải tìm trên giá trị trên cả 4 lươt quay, so sánh và xác định giá trị lớn nhất nằm ở lượt quay nào. (Vì mỗi lượt quay có 1 giá trị khác nhau, và chúng không liên quan đến nhau) -> Phải lưu trữ 4 giá trị, và thực hiện so sánh tìm giá trị lớn nhất trong 4 số -> Sử dụng mảng để lưu trữ và truy xuất giá trị từng lượt quay qua chỉ số trong mảng sẽ tốt nhất -> **Giải quyết được vấn đề lưu trữ**
* a,b,c,d là các số có điều kiện, đòi hỏi người dùng phải xử lý điều kiện này trước khi vào giải bài toán -> **Giải quyết được vấn đề trường hợp** (các trường hợp không thực hiện được và thực hiện được)
* Sử dụng thuật toán tìm kiếm và lưu trữ trong mảng để ghi và tìm giá trị cũng như số lần quay -> **giải quyết vấn đề thuật toán**
* Bài tập này có thể tổ chức theo hàm, vì việc tính chỉ số trên từng lượt quay được thực hiện giống nhau, nếu viết đi viết lại sẽ rất mất thời gian -> **giải quyết vấn đề tổ chức chương trình**

Bài làm:

#include <iostream>

using namespace std;

float r\_value(int a,int b,int c,int d){

return (float)a/c - (float)b/d;

}

int main(){

int a,b,c,d;

cin >> a >> b >> c >> d;

if (a < 0 || b < 0 || c < 0 || d < 0){

cout << “Du lieu khong hop le ! Thu lại !”;

cin >> a >> b >> c >> d;

}

float r[4];

r[0] = r\_value(a,b,c,d);

r[1] = r\_value(c,a,d,b);

r[2] = r\_value(d,c,b,a);

r[3] = r\_value(b,d,a,c);

int max = 0;

for (int i = 1; i <= 3; i++){

if (r[i] > r[max])

max = i;

}

cout << max;

return 0;

}

1. **Bài tập thực hành và chữa trên lớp:**

**Bài 1:** (Nguồn: Codelearn)

Trong toán học, việc tìm các số khi biết một số ràng buộc về số đó là rất phổ biến. Người ta thường diễn đạt các ràng buộc đó dưới dạng một đẳng thức (A=B). Khi đó, việc tính toán ra số cần tìm được gọi là giải phương trình hay tìm nghiệm của phương trình, số cần tìm gọi là ẩn. Một hệ phương trình sẽ gồm nhiều phương trình có liên quan đến nhau, thường có nhiều ẩn. Tương tự như giải phương trình, việc tìm ra các số phù hợp với bài toán gọi là giải hệ phương trình. Một hệ phương trình bậc nhất 2 ẩn thường có biểu diến như sau:



Trong đó a, b, m, n, k, q là các số cho trước, x và y là các ẩn hay các số cần tìm. Hệ phương trình có thể có 1 cặp nghiệm x,y, không có nghiệm hoặc có vô số nghiệm. Hãy thử giải quyết bài toán này. Nghiệm của hệ phương trình dưới dạng "x,y" nếu phương trình có 1 cặp x,y thỏa mãn. Nếu không có số thỏa mãn thì đầu ra là "Can not solve", nếu có vô số cặp thỏa mãn hãy trả về "Infinities". Kết quả có nhận được nếu là số thực thì làm tròn đến 1 chữ số thập phân.

**Ví dụ:**

* Với numbers={1, 1, 2, 1, 2, 8}, kết quả sẽ là setOEequations(numbers)= -4,6  
  Hệ phương trình sẽ có dạng:

  
  Ta có các số x=-4 và y=6 thỏa mãn hệ phương trình trên.

**Bài 2** (Nguồn: 28tech)

Tất cả các thành phố của Lineland đều nằm trên trục tọa độ Ox. Do đó, mỗi thành phố được liên kết với vị trí xi - tọa độ trên trục Ox. Không có hai thành phố được đặt tại một điểm.

Cư dân Lineland thích gửi thư cho nhau. Một người chỉ có thể gửi thư nếu người nhận sống ở một thành phố khác. Chi phí gửi thư chính xác bằng khoảng cách giữa thành phố của người gửi và thành phố của người nhận.

Đối với mỗi thành phố, hãy tính hai giá trị mini và maxi, trong đó mini là chi phí tối thiểu để gửi thư từ thành phố thứ i đến một thành phố khác và maxi là chi phí tối đa để gửi thư từ thành phố thứ i đến một số thành phố khác

Input

Dòng đầu tiên của đầu vào chứa số nguyên n (2<= n ≤ 10^5) - số lượng thành phố trong Lineland. Dòng thứ hai chứa chuỗi n số nguyên khác nhau x1, x2, ..., xn (- 10^9<= xi <=10^9), trong đó xi là tọa độ x của thành phố thứ i. Tất cả các xi là khác biệt và theo thứ tự tăng dần.

Output

In n dòng, dòng thứ i phải chứa hai số nguyên mini, maxi, cách nhau bởi một

khoảng trắng, trong đó mini là chi phí tối thiểu để gửi thư từ thành phố thứ i và maxi là chi phí tối đa để gửi thư từ thành phố thứ i.

**Bài 3:** Nguồn: Codelearn

Một **số gapful** là một số tự nhiên có ít nhất 3 chữ số sao nó cho chia hết cho số được tạo bởi chữ số đầu tiên và chữ số cuối cùng của nó. Cho số nguyên n, hãy kiểm tra xem **n** có phải là **số gapful** không

**Ví dụ:**

* Với n = 108 thì kết quả gapful\_number(n) = true  
  Giải thích:
  + 108 chia hết cho 18 nên 108 là số gapful.

**Bài 4: (**Nguồn: Codelearn)

Nam rất thích uống Coca. Trên đường đi học về cậu gặp một cửa hàng có chương trình đổi y vỏ chai lấy 1 chai nguyên. Nhưng cậu chỉ có tiền mua được x chai coca. Hãy giúp Nâm tính xem từ x chai nguyên ban đầu đó thì cậu sẽ được uống bao nhiêu chai coca.

**Ví dụ:**

* Với x = 15, y = 4 thì numCoca(x, y) = 19. Vì ban đầu Nâm có 15 chai coca. Sau khi uống hết 15 chai đó, Nâm đổi được thêm được 3 chai Coca mới và còn lại 3 vỏ chai. rồi Nâm đổi thêm được 1 chai Coca nữa và còn thừa lại 2 vỏ chai. Cuối cùng Nâm không đổi được nữa. Tổng cộng là 15 + 3 + 1 = 19.
* Với x  = 2, y = 3 thì numCoca(x, y) = 2. Nâm uống 2 chai coca xong thì không đổi được vỏ chai.

**Bài 5:** (Nguồn: Phần bài tập số 5 - Bài tập lớn lấy điểm thi kết thúc học phần môn Kỹ thuật lập trình C – Trường Đại học Khoa học tự nhiên Thành phố Hồ Chí Minh)

Hãng xe điện X đang tiến hành thử nghiệm hiệu suất pin của xe điện kiểu mới nhất do hãng đó chế tạo trên quãng đường đèo dốc. Hãy viết chương trình tính số km đã di chuyển trên thực tế khi sử dụng xe điện đó. Biết đoạn đường mà xe đang đi toàn đèo dốc, đoạn đường bằng có độ dài không đáng kể Giả sử rằng, ban đầu pin của xe được sạc đầy (100%)

Input: Chuỗi các số nguyên thể hiện hành trình, trong đó.

+ Số nguyên dương thể hiện xe đang lên dốc (với 2,15km/1% pin)

+ Số 0 thể hiện xe dừng sạc pin. Mỗi lần dừng sạc pin tăng 25% dung lượng

+ Số âm thể hiện xe đang xuống dốc, khi đó pin được sạc bù lại. Cứ đủ 10km xuống dốc pin sẽ tăng thêm 1%

Mức pin của mỗi giai đoạn luôn được làm tròn thành số nguyên

Output: Số km đã đi được trên thực tế (giá trị được làm tròn đến 1 chữ số thập phân.

Lưu ý: Nếu pin xe sạc khi đã đầy thì dung lượng tối đa là 100% và xe dừng lại nếu dung lượng pin là 0% hoặc hết hành trình

Bài 6: (Nguồn: LuyenCode) Tháng 7 năm nay trường ta (THPT Chuyên Sơn La) đăng cai tổ chức THHV lần thứ XV. Để chuẩn bị cho sự kiện trọng đại này, nhà trường cần sự tham gia tích cực của các bạn học sinh vào công tác tình nguyện (tiếp đón đại biểu các trường, chỉ dẫn đường đi, …).

Để chọn đội tình nguyện viên, BTC THHV muốn chọn những em học sinh có chiều cao bằng nhau (để được đội hình đồng đều, đẹp). Biết rằng nhà trường có n em học sinh có chiều cao lần lượt là ℎ1,ℎ2,…,ℎn

****Yêu cầu:**** Hãy giúp BTC THHV chọn ra một nhóm học sinh có chiều cao bằng nhau với số lượng nhiều nhất, nếu có nhiều nhóm có số lượng bằng nhau thì chọn nhóm có chiều cao lớn nhất

Bài 7: (Nguồn Luyencode)

Dr. Patel đang học tiếng Anh, chợt trong đầu nảy ra một ý tưởng về một ngôn ngữ cho riêng bản thân mình, Dr. Patel đã làm như sau.

Trong ngôn ngữ tiếng anh gồm các phụ âm và các nguyên âm, nguyên âm gồm các ký tự a, u, o, e và i, còn lại là phụ âm. Ngôn ngữ mới của Patel sẽ được chuyển từ tiếng Anh sang bằng cách tất cả các chữ cái thuộc nguyên âm sẽ đều nhân đôi ký tự của chúng.

Ví dụ: chuỗi ký tự ehlo finalist sẽ được chuyển thành eehloo fiinaaliist.

Dr. Patel đang tạo dựng từ điển nên cần một chương trình tốt để giúp việc chuyển đổi ngôn ngữ dễ dàng hơn. Cho một chuỗi ký tự � có độ dài � là chuỗi ký tự trong ngôn ngữ của Dr. Patel, hãy viết chương trình chuyển ngôn ngữ ấy về tiếng Anh.