ĐẠI HỌC BÁCH KHOA HÀ NỘI

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

----- 🙡 🕮 🙣 -----



**BÁO CÁO PROJECT I**

***Báo cáo tuần 10***

Giảng viên: **Ngô Lam Trung**

Sinh viên thực hiện: Nguyễn Quang Long

Mã số sinh viên: 20194320

**Hà Nội, năm 2022**

Nội dung đã hoàn thành:

Từ câu 26 đến câu 40

26. Even Digits Only

a. Đề bài

Kiểm tra xem tất cả các chữ số của số đã cho có phải số chẵn không

b. Giải pháp

Chuyển số đó thành mảng rồi xét từng phần tử

c. Code

bool solution(int n) {

    string n1 = to\_string(n);

    for(int i = 0; i<n1.size();i++){

        if((n1[i]-'0')%2!=0) return false;

    }

    return true;

}

27. Variable Name

a. Đề bài

Kiểm tra xem 1 tên biến có hợp lệ không (chỉ gồm các chữ cái tiếng Anh, chữ số và dấu gạch dưới, không thể bắt đầu bằng số)

b. Giải pháp

Kiểm tra từng ký tự trong tên biến xem có thỏa mãn điều kiện khôgn

c. Code

bool solution(string name) {

    if(isdigit(name[0])) return false;

    else for(char c : name) {

        if(!isdigit(c) && !isalpha(c) && c!='\_')

            return false;

    }

    return true;

}

28. Alphabetic Shift

a. Đề bài

Cho 1 xâu, thay thế các ký tự của nó bằng chữ cái tiếp theo trong bảng chữ cái alphabet

b. Giải pháp

Nếu ký tự đang xét là ‘z’ thì chuyển thành ‘a’, còn lại chuyển thành ký tự tiếp theo

c. Code

string solution(string inputString) {

    for(int i = 0; i<inputString.size();i++){

        if(inputString[i] == 'z') inputString[i]='a';

        else inputString[i]++;

    }

    return inputString;

}

29. Chess Board Cell Color

a. Đề bài

Cho vị trí 2 ô trong bàn cờ, kiểm tra xem chúng có cùng màu không

VD For cell1 = "A1" and cell2 = "C3", the output should be  
solution(cell1, cell2) = true.

A picture containing background pattern

Description automatically generated

b. Giải pháp

Nếu tổng vị trí trục ngang và trục dọc của cả 2 vị trí đã cho là số chẵn thì 2 ô cùng màu, ngược lại khác màu

c. Code

bool solution(string cell1, string cell2) {

    int x1 = cell1[0] -'A', y1 = cell1[1] - '1';

    int x2 = cell2[0] -'A', y2 = cell2[1] - '1';

    if((x1+x2+y1+y2)%2==0) return true;

    else return false;

}

30. Circle of Numbers

a. Đề bài

Cho các số nguyên từ 0 đến n-1 viết lần lượt theo vòng tròn với khoảng cách giữa 2 số liên tiếp bằng nhau (như hình dưới) Cho n và firstNumber, xác định xem số nào nằm ở phía đối diện với firstNumber

VD: n = 10, firstNumber = 2 => 7

A picture containing accessory, necklet

Description automatically generated

b. Giải pháp

Nhận thấy |kq – firstNumber| = n/2 => kq

c. Code

int solution(int n, int firstNumber) {

    n/=2;

    if(firstNumber>=n) return firstNumber-n;

    else return firstNumber+n;

}

31. Deposit Profit

a. Đề bài

Giả sử bạn gửi tiết kiệm vào ngân hàng 1 số tiền deposit với lãi suất rate, kiểm tra xem sau bao lâu thì số tiền gốc vượt qua threshold

b. Giải pháp

Đây là bài toán lãi kép: deposit\*(1+rate)^n >= threshold

c. Code

int solution(int deposit, int rate, int threshold) {

    int n =0;

    float increase = 1+rate/100.0;

    float cur = deposit;

    while(cur<threshold) {

        cur\*=increase;

        n++;

    }

    return n;

}

32. Absolute Values Sum Minimization

a. Đề bài

Cho 1 mảng các số nguyên đã được sắp xếp, xác định xem phần tử x nào tạo ra giá trí nhỏ nhất của biểu thức

abs(a[0] - x) + abs(a[1] - x) + ... + abs(a[a.length - 1] - x)

nếu có nhiều đáp án, đưa ra x nhỏ nhất

b. Giải pháp

Do mảng đã sắp xếp nên phần tử x sẽ là phần tử nằm giữa mảng nếu mảng có lẻ phần tử, nếu mảng có chẵn phần tử thì tính giá trị biểu thức đối với 2 phần tử nằm giữa rồi so sánh

c. Code

int solution(vector<int> a) {

    int len = a.size();

    if(len%2!=0) return a[len/2];

    else {

        int sum1=0, sum2=0;

        for(int i = 0; i<len ;i++ ){

         sum1 += abs(a[i]-a[len/2-1]);

         sum2 += abs(a[i]-a[len/2]);

            }

        if(sum1<=sum2) return a[len/2-1];

        else return a[len/2];

}

}

33. Strings Rearrangement

a. Đề bài

Cho 1 mảng các xâu có độ dài bằng nhau, kiểm tra xem có thể sắp xếp lại các xâu đó sao cho 2 xâu liên tiếp chỉ khác nhau 1 ký tự không?

b. Giải pháp

Duyệt tất cả các hoán vị có thể có của mảng và kiểm tra từng phần tử

c. Code

bool solution(std::vector<std::string> inputArray) {

    while(next\_permutation(inputArray.begin(), inputArray.end())) {

        bool flag = true;

        for(int i = 0; i < inputArray.size() - 1; i++) {

            int diff = 0;

            for(int j = 0; j < inputArray[i].size(); j++)

                if(inputArray[i][j] != inputArray[i+1][j]) diff++;

            if(diff != 1) flag = false;

        }

        if(flag) return true;

    }

    return false;

}

34. Extract Each Kth

a. Đề bài

Cho 1 mảng các số nguyên, hãy loại bỏ phần tử đang xét sau mỗi k phần tử khi duyệt mảng

b. Giải pháp

Duyệt mảng, nếu phần tử đang xét chia hết cho k thì không lấy, ngược lại thì lấy

c. Code

vector<int> solution(vector<int> inputArray, int k) {

    vector<int> res ;

    for(int i =0 ; i<inputArray.size();i++){

       if((i+1)%k!=0) res.push\_back(inputArray[i]);

    }

    return res;

}

35. First Digit

a. Đề bài

Tìm chữ số trái nhất xuất hiện trong xâu cho trước (xâu chứa ít nhất 1 số)

b. Giải pháp

Duyệt lần lượt các ký tự, nếu gặp ký tự là số thì trả về ký tự đó

c. Code

char solution(string inputString) {

    for(char c: inputString){

        if(isdigit(c)) return c;

    }

}

36. Different Symbols Naïve

a. Đề bài

Cho 1 xâu, tìm số ký tự khác nhau của xâu đó

b. Giải pháp

Sử dụng set push các ký tự của xâu rồi trả về kích thước của set

c. Code

int solution(string s) {

    set<char> res;

    for(char c: s) {

        res.insert(c);

    }

    return res.size();

}

37. Array max Consecutive Sum

a. Đề bài

Cho 1 mảng các số nguyên, tìm tổng lớn nhất của k phần tử liên tiếp

b. Giải pháp

Xét tất cả các mảng con rồi tính tổng

c. Code

int solution(vector<int> inputArray, int k) {

    int max =0;

    for(int i=0; i<=inputArray.size()-k;i++){

        int j = i;

        int sum =0;

        for(int j1 = 0; j1<k ;j1++){

            sum+=inputArray[j+j1];

        }

        if(max < sum) max =sum;

    }

    return max;

}

38. Growing Plant

a. Đề bài

Tính xem mất bao nhiêu ngày để 1 cái cây đạt được 1 chiều cao biết ban ngày cây tăng x ban đêm giảm y (chiều cao ban đầu là 0)

b. Giải pháp

Tăng và giảm chiều cao lần lượt đến khi đạt giá trị cần tìm

c. Code

int solution(int upSpeed, int downSpeed, int desiredHeight) {

    int cur = 0, i=0;

    do{

        i++;

        cur+= upSpeed;

        if(cur>=desiredHeight) break;

        cur-=downSpeed;

    } while(cur<desiredHeight);

    return i;

}

39. Knapsnack Knight

a. Đề bài

Có 2 items, item1 nặng weight1, có giá trị value1, item2 nặng weitght2, có giá trị value2, bạn chỉ có thể mang được trọng lượng tối đa maxW. Hỏi giá trị lớn nhất có thể mang là bao nhiêu

b. Giải pháp

Nếu maxW > tổng 2 item thì trả về value1+value2, nếu maxW < min 2 item thì trả về 0. Nếu maxW >= max 2 item thì trả về giá trị lớn nhất, ngược lại trả về value của item có weight nhỏ hơn

c. Code

int solution(int value1, int weight1, int value2, int weight2, int maxW) {

    if(maxW>=weight1+weight2) return value1+value2;

    if(maxW<min(weight1,weight2)) return 0;

    if(maxW>=max(weight2,weight1)) return max(value2,value1);

    else if(weight1>weight2) return value2;

    else return value1;

}

40. Longest Digits Prefix

a. Đề bài

Cho 1 xâu, đưa ra tiền tố dài nhất của xâu đó (chỉ chứa chữ số)

b. Giải pháp

Duyệt từ trái sang phải, nếu là chữ số thì thêm vào xâu, nếu không phải thì dừng vòng lặp

c. Code

string solution(string inputString) {

    string res = "";

    for(int i =0; i<inputString.size();i++){

        if(isdigit(inputString[i])) res+=inputString[i];

        else break;

    }

    return res;

}