

MỤC LỤC

PHẦN I - CẤU TRÚC RỄ NHÁNH 4

 Bài 0: Tính diện tích, chu vi các hình cơ bản 4

 Bài 1: Tìm giá trị lớn nhất, nhỏ nhất của 3 số a, b, c. 5

 Bài 2: Tìm nghiệm của phương trình: $ax + b = 0$ 5

 Bài 3: Tìm nghiệm của phương trình: $ax^2 + bx + c = 0$ 5

 Bài 4: Tìm nghiệm của Hệ phương trình: $\begin{cases} ax + by = c \\ a'x + b'y = c' \end{cases}$ 5

 Bài 5: Nhập vào một tháng và cho biết số ngày của tháng đó. 7

 Bài 6: Nhập vào một số có 3 chữ số $\overline{abc} \in Z^+$, sau đó hiện ra màn hình chữ số hàng trăm, chục, đơn vị. 7

 Bài 7: Nhập vào một số có 3 chữ số $\overline{abc} \in Z^+$, hiện ra màn hình số đảo ngược (xem bài 6) 8

 Bài 8: Tính $\sqrt[n]{X}$ 8

PHẦN 2 - CẤU TRÚC LẶP..... 9

 Bài 1: Nhập một số $N \in Z^+$ kiểm tra xem số đó có là số nguyên tố hay không. 9

 Bài 2: Nhập một số $N \in Z^+$ hiện các số chính phương từ $1 \rightarrow N$ 10

 Bài 3: Đếm số ước chẵn, lẻ của N. 10

 Bài 4: Lập chương trình nhập vào một số nguyên dương N ($n \leq 10$). Sau đó hiện ra màn hình tam giác chứa các kí tự '*'. 10

 Bài 5: Lập chương trình nhập số $N \in Z^+$, đếm số chữ số của N. 11

 Bài 6: $S = 1 + 3 + 5 + \dots + n (n \in Z^+)$ 11

 Bài 7: $S = 1 * 1 + 2 * 2 + 3 * 3 + \dots + n * n (n \in Z^+)$ 11

 Bài 8: $S = \frac{1}{3} + \frac{2}{4} + \frac{3}{5} + \dots + \frac{n}{n+2} (n \in Z^+)$ 12

 Bài 9: $S = \frac{1}{2} - \frac{2}{3} + \frac{3}{4} - \frac{4}{5} + \dots + (-1)^{n+1} \frac{n}{n+1} (n \in Z^+)$ 12

 Bài 10: $P = 1 \times 3 \times 5 \times \dots \times N (N \in Z^+)$ 13

 Bài 11: Tính tổng các ước chẵn, ước lẻ của N ($N \in Z^+$) (xem Bài 3). 13

 Bài 13: Đếm số lượng số nguyên tố bé hơn N, với $N \in Z^+$ nhập từ bàn phím (xem Bài 1) 13

 Bài 14: Kiểm tra một số $N \in Z^+$ có đối xứng gương hay không. 13

 Bài 15: Tìm Ước chung lớn nhất UCLN của 2 số nguyên dương a,b. 13

 Bài 16: Tìm Bội chung nhỏ nhất BCNN của 2 số nguyên dương a,b. 14

 Bài 17: Tối giản phân số $\frac{a}{b}$, với $a, b \in Z^+$ lần lượt là tử số và mẫu số nhập từ bàn phím. 14

 Bài 18: Nhập hai số nguyên dương a,b. Kiểm tra xem chúng có nguyên tố cùng nhau không? 14

Bài 19: Lập chương trình nhập tuổi bố (tb), tuổi con (tc) (hiện tại $tb > 2tc$, $tb - tc \geq 25$). Hiện ra mà hình câu trả lời “ <i>Bao nhiêu năm nữa thì tuổi bố gấp đôi tuổi con</i> ”.....	14
Bài 20: Tìm K min sao cho: $2^k > N (N \in \mathbb{Z}^+)$, N nhập từ bàn phím.....	15
Bài 21: Nhập và tính tổng các số nguyên nhập từ bàn phím cho đến khi giá trị nhập vào bằng 0 thì dừng lại và thông báo tổng tính được.	15
Bài 22: Một người gửi tiết kiệm không kỳ hạn với số tiền G VND với lãi suất 0.0083% mỗi tháng. Hỏi sau bao nhiêu tháng thì có thể rút số tiền lớn hơn hoặc bằng R VND.	15
Bài 23: Phân tích số $N \in \mathbb{Z}^+$ thành tích các thừa số nguyên tố, N nhập từ bàn phím.....	16
Bài 24: MENU tự tạo.....	16
Bài 25: Tìm và in số hoàn hảo.....	17
PHẦN 3 - MẢNG MỘT CHIỀU	19
Bài 0: Khai báo, nhập, xuất mảng 1 chiều A.....	19
Bài 1: Tìm Max, Min của M1C A.....	19
Bài 2: Tìm UCLN của M1C A.....	20
Bài 3: Sắp xếp M1C A thành dãy không giảm (<i>Nổi bọt, tráo đổi</i>).	20
Bài 4: M1C A đã sắp xếp không giảm, nhập số nguyên K. Chèn K vào M1C A sao cho M1C A vẫn giữ được tính chất không giảm.....	21
Bài 5: Nhập số nguyên K. Tìm xem K có xuất hiện trong dãy A không, nếu có thông báo vị trí i sao cho $A_i = K$	22
Bài 6: Tìm phần tử xuất hiện nhiều lần nhất trong M1C A.	22
Bài 7: Tính tổng các phần tử chia hết cho k (<i>k nhập từ bàn phím</i>).	23
Bài 8: Tìm phần tử lớn nhì trong M1C A.	23
Bài 9: Hiện ra màn hình các phần tử là số nguyên tố trong M1C A.....	23
Bài 10: Hiện ra màn hình các số chính phương trong M1C A.	24
Bài 11: Kiểm tra dãy A có là cấp số cộng (CSC) hay không, nếu có đưa ra công sai (d).	24
Bài 12: Trộn hai mảng đã được sắp xếp theo thứ tự <i>tăng dần</i> lại thành một mảng mới mà vẫn đảm bảo thứ tự của nó. (<i>Chú ý: không dùng thuật toán sắp xếp</i>).....	24
Bài 13: Tìm dãy số nguyên dương liên tiếp, không giảm, dài nhất trong mảng, ghi ra chỉ số đầu tiên và cuối cùng của dãy tìm được.	25
Bài 14: Tìm số phần tử nhỏ hơn phần tử tại vị trí thứ i trong mảng.....	26
Bài 15: Tìm tổng lớn nhất của K phần tử liên tiếp trong dãy.	26
Bài 16: Tìm số Fibonaxi lớn nhất trong một dãy gồm N số nguyên dương.	27
PHẦN 4 - KIỂU XÂU (~10⁸ kí tự)	29
Bài 1: Kiểm tra xâu đó có đối xứng gương hay không.....	29
Bài 2: Đếm số kí tự là chữ số trong xâu.....	29
Bài 3: Đếm số kí tự là nguyên âm trong xâu. (<i>Các chữ cái là nguyên âm: UEOAI</i>).....	29
Bài 4: Hiện ra màn hình xâu đó dạng chữ in hoa & in thường.....	30
Bài 5: Hiện ra màn hình mỗi từ trên một dòng.....	30
Bài 6: Tính tổng tất cả các chữ số trong xâu.....	30

Bài 7: Đếm số từ có trong xâu.	31
Bài 8: Chuẩn hóa lại xâu vừa nhập.	31
Bài 9: Xâu nhập vào có dạng dãy bit (0,1) Đưa bit 1 về đầu xâu, bit 0 về cuối xâu.	31
Bài 10: Cho biết kí tự xuất hiện nhiều lần nhất trong xâu.	32
Bài 11: Nhập xâu cần tìm s1, nhập xâu cần thay thế s2. Tìm và thay thế tất cả các xâu s1 bằng xâu s2. ...	32
Bài 12: Nhập vào khóa K, hãy mã hóa xâu vừa nhập theo phương pháp mã hóa Ceda:	33
Bài 13: Nhập vào một năm dương lịch N, đổi sang năm âm lịch tương ứng.	34
Bài 14: Nhập vào một xâu chứa các chữ số, kiểm tra xem xâu vừa nhập có là SỐ không?	34
Bài 15: Nhập vào một số nguyên dương N, đọc số đó ($0 < N \leq 10^{11}$)	34
Bài 16: Nhập vào 2 số nguyên lớn (tối đa 500 chữ số). Tính Tổng hai số vừa nhập.	36
Bài 17: Nhập vào 2 số nguyên lớn (tối đa 500 chữ số). Tính Hiệu hai số nguyên vừa nhập.	37
Bài 18: Nhập vào 2 số nguyên lớn (tối đa 500 chữ số). Tính Tích hai số nguyên vừa nhập.....	38
Bài 19: Đổi 1 số hệ 10 sang hệ 2 và ngược lại.....	41
Bài 20: Đổi 1 số hệ 10 sang hệ 16 và ngược lại.....	41
Bài 21: Đổi 1 số hệ 16 sang hệ 2 và ngược lại.....	41

PHẦN I - CẤU TRÚC RỄ NHÁNH

Bài 0: Tính diện tích, chu vi các hình cơ bản

```
//#include <iostream>
#include <cmath>
#include<bits/stdc++.h>
const float pi=3.14;
using namespace std;
// Diện tích và chu vi hình tròn bán kính R
void SC_HinhTron(){
    float s,c,r;
    cout << "Nhap R="; //<<endl;
    cin >> r;
    if (r>0){
        s=pi*r*r;
        c=pi*r;
        cout << "Diện tích hình tròn là:"<<setprecision(2)<<fixed<<s;
        cout << "\nChu vi hình tròn là:"<<setprecision(2)<<fixed<<c;
    }
    else cout <<"Nhap lại R";
}
// Diện tích và chu vi hình chữ nhật
void SC_HinhCN(){
    float s,c,a,b;
    cout<<"Nhap chiều dài a="; cin>>a;
    cout<<"Nhap chiều rộng b="; cin>>b;
    if(a>0 && b>0){
        s=a*b;
        c=(a+b)*2;
        cout<<"\nDiện tích HCN là:"<<setprecision(2)<<fixed<<s;
        cout<<"\nChu vi HCN là:"<<setprecision(2)<<fixed<<c;
    }
    else cout<<"Nhap sai!";
}
//Diện tích Tam giác Herong
void SC_TamGiac(){
    float s,cv,p,a,b,c;
    cout<<"Nhap cạnh a="; cin>>a;
    cout<<"Nhap cạnh b="; cin>>b;
    cout<<"Nhap cạnh c="; cin>>c;
    if (a>0 && b>0 && c>0 && a+b>c && b+c>a && a+c>b){
        cv=(a+b+c);
        p=cv/2;
        s=sqrt(p*(p-a)*(p-b)*(p-c));
        cout<<"\nDiện tích tam giác là:"<<setprecision(2)<<fixed<<s;
        cout<<"\nChu vi tam giác là:"<<setprecision(2)<<fixed<<cv;
    }
    else cout<<"a,b,c không là cạnh tam giác!";
}
int main(){
    //SC_HinhTron();
    //SC_HinhCN();
}
```

```

    SC_TamGiac();
    return 0;}
/*Ghi chu:
- Ham: setprecision(n)<<fixed<<x: lam tron so thuc x den n chu so thap
phan
*/

```

Bài 1: Tìm giá trị lớn nhất, nhỏ nhất của 3 số a, b, c.

```

#include<bits/stdc++.h> //Chứa hầu hết các thu viện.
using namespace std;
int main(){
    int a,b,c;
    cout<<"Nhap a="; cin>>a;
    cout<<"Nhap b="; cin>>b;
    cout<<"Nhap c="; cin>>c;
    /* Cach 1
    if (a>b && a>c) cout<<"Max="<<a;
    if (b>a && b>c) cout<<"Max="<<b;
    if (c>a && c>b) cout<<"Max="<<c;
    */
    /* Cach 2
    int mymax=a;
    if (mymax<b) mymax=b;
    if (mymax<c) mymax=c;
    cout<<"Max="<<mymax;
    */
    int mymax=max(a,max(b,c));
    cout<<"Max="<<mymax;
    int mymin=min(a,min(b,c));
    cout<<"\nMin="<<mymin;
return 0;}

```

Bài 2: Tìm nghiệm của phương trình: $ax + b = 0$.

Bài 3: Tìm nghiệm của phương trình: $ax^2 + bx + c = 0$.

Bài 4: Tìm nghiệm của Hệ phương trình: $$\begin{cases} ax+by=c \\ a'x+b'y=c' \end{cases}$$

```

#include<bits/stdc++.h>
using namespace std;

void GiaiPT_bac1(){
float a,b,x;
cout<<"Nhap he so a="; cin>>a;
cout<<"Nhap he so b="; cin>>b;
if (a!=0)
    cout<<"\nX="<<setprecision(2)<<fixed<<-b/a;
else if(b==0)
    cout<<"\nPT vo so nghiem!";
else
    cout<<"\nPT vo nghiem!";
}

void GiaiPT_bac2(){
float a,b,c,x1,x2,delta;

```

```

cout<<"Nhap he so a="; cin>>a;
cout<<"Nhap he so b="; cin>>b;
cout<<"Nhap he so c="; cin>>c;
    delta=b*b-4*a*c;
    if(delta>0){
        x1=(-b-sqrt(delta))/(2*a);
        x2=(-b+sqrt(delta))/(2*a);
        cout<<"X1="<<setprecision(2)<<fixed<<x1<<endl;
        cout<<"X2="<<setprecision(2)<<fixed<<x2<<endl;
    }else if (delta==0)
        cout<<"PT co nghiem kep x1=x2="<<setprecision(2)<<fixed<<-b/(2*a);
    else cout<<"PT Vo nghiem!";
}
/*
Cho He PT:
a1x+b1y=c1
a2x+b2y=c2
Sử dụng định thức Cramer!
Tinh D, Dx, Dy --> x=Dx/D; y=Dy/D.
*/

void Giai_HePT(){
float a1,b1,c1,a2,b2,c2,D,Dx,Dy;
cout<<"Nhap he so PT1 a1,b1,c1="; cin>>a1>>b1>>c1;
cout<<"Nhap he so PT2 a2,b2,c2="; cin>>a2>>b2>>c2;
D=a1*b2-a2*b1;
Dx=b2*c1-b1*c2;
Dy=a1*c2-a2*c1;

if(D!=0)cout<<"He PT co nghiem
("<<setprecision(2)<<fixed<<Dx/D<<","<<setprecision(2)<<fixed<<Dy/D<<")";
else if(Dx==0 && Dy==0) cout<<"He PT co vo so nghiem!";
    else cout<<"He PT vo nghiem!";
}

int main(){
    //GiaiPT_bac1();
    //GiaiPT_bac2();
    Giai_HePT();
return 0;}

```

Bài 5: Nhập vào một tháng và cho biết số ngày của tháng đó.

- Với $t = 1; 3; 5; 7; 8; 10; 12$ (Tháng có 31 ngày)
- Với $t = 4; 6; 9; 11$ (Tháng có 30 ngày)
- Với $t = 2$ Nhập năm, nếu năm nhuận thì có 29 ngày, còn lại 28 ngày.

(N là năm nhuận khi thoả mãn điều kiện: $(N\%400==0) || ((N\%4==0 \ \&\& \ N\%100!=0))$)

```

/*
Nhập vào một tháng và cho biết số ngày của tháng đó.
- Với t = 1; 3; 5; 7; 8; 10; 12 (Tháng có 31 ngày)
- Với t = 4; 6; 9; 11 (Tháng có 30 ngày)
- Với t = 2 Nhập năm, nếu năm nhuận thì có 29 ngày, còn lại 28 ngày.
*/
#include<bits/stdc++.h>
using namespace std;

int main() {
    int m,kl,dm2;
    cout<<"Nhập vào một tháng m="; cin>>m;
    //Nếu tháng 2 thì nhập năm, kiểm tra nếu năm nhuận thì tháng 2 có
    29 ngày, ngược lại 28 ngày
    if (m==2) {
        int n;
        cout<<"Bạn hãy nhập năm dương lịch N=";
        cin>>n;
        if(n%400==0 || (n%100!=0 && n%4==0)) dm2=29; else dm2=28;
    }
    kl=(m==1 || m==3 || m==5 || m==7 || m==8 || m==10 || m==12) ? 31 : (m==4 || m==6 || m==9 ||
    m==11) ? 30 : (m==2) ? dm2 : 0;
    if (kl==0) cout<<"Thang nhap vào không tồn tại!";
    else cout<<"Thang "<<m<<" có "<<kl<<" ngày!";
    cout<<endl;
return 0; }

```

Bài 6: Nhập vào một số có 3 chữ số $\overline{abc} \in \mathbb{Z}^+$, sau đó hiện ra màn hình chữ số hàng trăm, chục, đơn vị.

```

#include<iostream>;
using namespace std;
/*Chương trình nhập 1 số nguyên có 3 chữ số;
Hiện ra các chữ số hàng trăm, chục, đơn vị */
int main() {
    int n=0;
    int tr,ch,dv=0;
    cout<<"Nhập số nguyên dương N=";
    cin>>n;
    if (n>=100 && n<=999) {
        tr=n/100;
        dv=n % 10;

```

```

    ch=(n %100)/10;
    cout<<"Chu so hang Tram la:"<<tr;
    cout<<"\nChu so hang Chuc la:"<<ch;
    cout<<"\nChu so hang Donvi la:"<<dv;
} else{ cout<<"So N khong la so co 3 chu so!";}
return 0;}

```

Bài 7: Nhập vào một số có 3 chữ số $\overline{abc} \in \mathbb{Z}^+$, hiện ra màn hình số đảo ngược (xem bài 6)

Bài 8: Tính $\sqrt[n]{X}$

```

#include<bits/stdc++.h>
#include<windows.h>
using namespace std;
//Tính căn bậc n của X;
int main(){
int n;
double kq,X;
bool ok;
system("color 4"); //Đặt màu con trỏ màu đỏ!
cout<<"\nNhap so can khai can X="; cin>>X;
cout<<"\nNhap so bac can khai can n="; cin>>n;
Sleep(3000);
system("color 7");
ok=(n<=0|| (n%2==0 && X<0))? false:true;
if(ok){
double bn=1.0/n;
kq=(X<0)? -pow(-X,bn):pow(X,bn);
cout<<"Can bac "<<n<<" của "<<X<<" = "<<setprecision(2)<<fixed<<kq;
} else cout<<"Khong tinh duoc!";
return 0; }

```


PHẦN 2 - CẤU TRÚC LẶP

Bài 1: Nhập một số $N \in \mathbb{Z}^+$ kiểm tra xem số đó có là số nguyên tố hay không.

```
#include<bits/stdc++.h>
using namespace std;
/*
bool SNT(int x){
//bool ok=true;
if (x==1) return false;
if (x==2 || x==3) return true;
else{
for(int i=2;i<=floor(sqrt(x));i++){
if(x%i==0) return false;
}
return true;
}
}
*/
bool SNT(int x){
bool isnt=true;
for(int i=2;i<=floor(sqrt(x));i++){
if(x%i==0)
{ isnt=false;
break;
}
}
bool ok = (x==1)?false:(x==2 || x==3)?true:isnt;
return ok;
}
int main(){
int n=100;
if (SNT(n)) cout<<"Dung";
else cout<<"Sai";
return 0;}
```

```
#include<bits/stdc++.h>
using namespace std;
int tt;
void laSNT(int x,int kq){
kq=1;
if(x==1) kq=0;
if(x==2 || x==3) kq=1;
else {
for(int i=2;i<=floor(sqrt(x));i++)
if (x%i==0) kq=0;}
}
int main(){
int n=10;
for (int k=1;k<=n;k++){
laSNT(k,tt); if(tt==1) cout<<k<<" ";
}
return 0;}
```

Bài 2: Nhập một số $N \in \mathbb{Z}^+$ hiện các số chính phương từ 1 \rightarrow N.

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int n;
    cout<<"\nNhap so nguyen duong N=";
    cin>>n;
    cout<<"\nCac so chinh phuong tu 1->"<<n<<" la:";
    for(int i=1;i<=n;i++){
        int t=sqrt(i);
        if (t*t==i) cout<<i<<" ";
    }
    return 0;}
```

Bài 3: Đếm số ước chẵn, lẻ của N.

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int n,uc=0,ul=0;
    cout<<"\nNhap so nguyen duong N=";
    cin>>n;
    for(int i=1;i<=n;i++){
        if(n%i==0 && i%2==0) uc++;
        if(n%i==0 && i%2!=0) ul++;
    }
    cout<<"\nSo uoc chan cua "<<n<<" la:"<<uc;
    cout<<"\nSo uoc le cua "<<n<<" la:"<<ul;
    return 0;}
```

Bài 4: Lập chương trình nhập vào một số nguyên dương N ($n \leq 10$). Sau đó hiện ra màn hình tam giác chứa các ký tự '*'.

N=3	N=5
*	*
***	***
*****	*****


```
#include<bits/stdc++.h>
using namespace std;
int main(){
    //ios_base::sync_with_stdio(false); //Chua quan tâm ☺
    //cin.tie(0);cout.tie(0); //Chua quan tâm ☺
    int h;
    cout<<"Nhap chieu cau h=";
    cin>>h;
```

```

    for(int i=1;i<=h;i++){
        for (int j=1;j<=2*i-1;j++) cout<<"*";
        cout<<endl;
    }
return 0;}

```

Bài 5: Lập chương trình nhập số $N \in \mathbb{Z}^+$, đếm số chữ số của N.

```

#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
//#define ll long long // Cách khác định nghĩa ll kiểu long long
int main(){
    ll n,n0;
    cout<<"\nBan nhap vao mot so nguyen N=";
    cin>>n;
    n0=n; //Biến n0 để lưu lại số n ban đầu.
    int dem=0;
    while(n>0){
        n/=10;
        dem++;
    }
    cout<<"\nSo chu so cua "<<n0<<" la:"<<dem;
return 0;}

```

Bài 6: $S = 1+3+5+\dots+n (n \in \mathbb{Z}^+)$

```

#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
//#define ll long long // Cách khác định nghĩa ll kiểu long long
int main(){
    ll n,s=0;
    cout<<"\nBan nhap vao mot so nguyen N=";
    cin>>n;
    // Dùng For...
    for(ll i=1;i<=n;i+=2) s+=i;
    // Dùng While
    /*ll i=1;
    while(i<=n){
        s+=i;
        i+=2; //i=i+2, số lẻ mà
    }*/
    cout<<"\nTong 1+3+5+...+ "<<n<<" = "<<s;
return 0;}

```

Bài 7: $S = 1*1+2*2+3*3+\dots+n*n (n \in \mathbb{Z}^+)$

```

#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
//#define ll long long // Cách khác định nghĩa ll kiểu long long
int main(){
    ll n,s=0;

```

```

cout<<"\nBan nhap vao mot so nguyen N=";
cin>>n;
// Dùng For...
// for(ll i=1;i<=n;i++) s+=i*i;
// Dùng While
ll i=1;
while(i<=n){
    s+=i*i;
    i++;
}
cout<<"\nTong 1*1+2*2+3*3+...+ "<<n<<"*"<<n<<" = "<<s;
return 0;}

```

Bài 8: $S = \frac{1}{3} + \frac{2}{4} + \frac{3}{5} + \dots + \frac{n}{n+2} (n \in \mathbb{Z}^+)$

```

#include<bits/stdc++.h>
using namespace std;
int main(){
    int n;
    double s=0;
    cout<<"\nBan nhap vao mot so nguyen N=";
    cin>>n;
    for(int i=1;i<=n;i++) s+=double(i)/(i+2);
    cout<<"\nTong 1/3+2/4+3/5+...+ "<<n<<"/"<<n+2<<" =
"<<setprecision(2)<<fixed<<s;
return 0;}

```

Bài 9: $S = \frac{1}{2} - \frac{2}{3} + \frac{3}{4} - \frac{4}{5} + \dots + (-1)^{n+1} \frac{n}{n+1} (n \in \mathbb{Z}^+)$

```

#include<bits/stdc++.h>
using namespace std;
int main(){
    int n;
    double s=0;
    cout<<"\nBan nhap vao mot so nguyen N=";
    cin>>n;
    // Cách 1
    /* for(int i=1;i<=n;i++)
        if(i%2!=0) s+=double(i)/(i+1);
        else s+=double(-i)/(i+1);*/
    // Cách 2
    for(int i=1;i<=n;i++) s+=double(i%2!=0? i:-i)/(i+1);
    //i lẻ là i, chẵn -i
    cout<<"\nTong 1/2-2/3+3/4+...+(-1)^(n+1) "<<n<<"/"<<n+1<<" =
"<<setprecision(3)<<fixed<<s;
return 0;}

```

Bài 10: $P = 1 \times 3 \times 5 \times \dots \times N (N \in \mathbb{Z}^+)$

```
#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
//#define ll long long // Cách khác định nghĩa ll kiểu long long
int main() {
    ll n,p=1;
    cout<<"\nBan nhap vao mot so nguyen N=";
    cin>>n;
    for(ll i=1;i<=n;i+=2) p*=i;
    cout<<"\nTich 1*3*5*...* "<<n<<" = "<<p;
return 0;}
```

Bài 11: Tính tổng các ước chẵn, ước lẻ của N ($N \in \mathbb{Z}^+$) (xem Bài 3)**Bài 13: Đếm số lượng số nguyên tố bé hơn N, với $N \in \mathbb{Z}^+$ nhập từ bàn phím (xem Bài 1)****Bài 14: Kiểm tra một số $N \in \mathbb{Z}^+$ có đối xứng gương hay không.**

```
#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
//#define ll long long // Cách khác định nghĩa ll kiểu long long
int main() {
    ll n1=1233210; ll n0=n1; ll n2=0;
    while(n1>0) {
        int x=n1%10;
        n2= n2*10+x;
        n1/=10; }
    if(n0==n2) cout<<"DXG"; else cout<<"Not DXG";
return 0;}
```

Bài 15: Tìm Ước chung lớn nhất UCLN của 2 số nguyên dương a,b.

```
#include<bits/stdc++.h>
using namespace std;
int main() {
    int a,b,x,y;
    cout<<"\nBan so nguyen duong a=";
    cin>>a;
    cout<<"\nBan so nguyen duong b=";
    cin>>b;
    x=a; y=b;
    while(x!=y)
        if(x>y) x-=y; //Số lớn = số lớn - số nhỏ
        else y-=x;
    cout<<"\nUCLN ("<<a<<","<<b<<")="<<x;
return 0;}
```

```
#include<bits/stdc++.h>
using namespace std;
int UCLN(int x,int y) {
    while(x!=y)
        if(x>y) x-=y;
```

```

        else y-=x;
    return x;
}
int main(){
    int a,b;
    cout<<"\nBan so nguyen duong a="; cin>>a;
    cout<<"\nBan so nguyen duong b="; cin>>b;
    cout<<"\nUCLN ("<<a<<" , "<<b<<" )="<<UCLN (a,b) ;
return 0;}

```

Bài 16: Tìm Bội chung nhỏ nhất BCNN của 2 số nguyên dương a,b.

```

#include<bits/stdc++.h>
using namespace std;
int UCLN(int x,int y){
    while(x!=y)
        if(x>y) x-=y;
        else y-=x;
    return x;
}
int main(){
    int a,b;
    cout<<"\nBan so nguyen duong a="; cin>>a;
    cout<<"\nBan so nguyen duong b="; cin>>b;
    cout<<"\nBCNN ("<<a<<" , "<<b<<" )="<<(a*b)/UCLN (a,b) ;
return 0;}

```

Bài 17: Tối giản phân số $\frac{a}{b}$, với $a, b \in \mathbb{Z}^+$ lần lượt là tử số và mẫu số nhập từ bàn phím.

```

#include<bits/stdc++.h>
using namespace std;
int UCLN(int x,int y){
    while(x!=y)
        if(x>y) x-=y;
        else y-=x;
    return x;
}
int main(){
    int ts,ms;
    cout<<"\nNhap Tu so="; cin>>ts;
    cout<<"Nhap Mau so="; cin>>ms;
    int d=UCLN(ts,ms);
    cout<<ts<<"/"<<ms<<" toi gian la: "<<ts/d<<"/"<<ms/d;
return 0;}

```

Bài 18: Nhập hai số nguyên dương a,b. Kiểm tra xem chúng có nguyên tố cùng nhau không?

Xem bài tìm UCLN: Tìm $UCLN(a,b)=1 \rightarrow a,b$ nguyên tố cùng nhau!

Bài 19: Lập chương trình nhập tuổi bố (tb), tuổi con (tc) (hiện tại $tb > 2tc$, $tb - tc = 25$). Hiện ra mà hình câu trả lời "Bao nhiêu năm nữa thì tuổi bố gấp đôi tuổi con".

```

#include<bits/stdc++.h>
using namespace std;
int main(){

```

```

int tb,tc;
do{
    cout<<"\nNhap Tuoi bo="; cin>>tb;
    cout<<"Nhap Tuoi con="; cin>>tc;
} while(tb<=2*tc & tb-tc<25); //Nhap lai khi tuoi bo, tuoi con vi phạm đk

int n=0;
while(tb!=2*tc){
    tb++;
    tc++;
    n++;
}
cout<<"\nSau "<<n<<" nam nua Tuoi bo = 2 lan Tuoi con!";
return 0;}

```

Bài 20: Tìm K min sao cho: $2^k > N (N \in \mathbb{Z}^+)$, N nhập từ bàn phím.

```

/* Tìm K min sao cho:  $2^k > N$ , N nhập từ bàn phím */
#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
int main(){
    ll N,K=0,T=1;
    cout<<"\nNhap so nguyen duong N="; cin>>N;
    while(T<=N){
        K++; T*=2;
    }
    cout<<"\nSo K min can tim la:"<<K;
return 0;}

```

Bài 21: Nhập và tính tổng các số nguyên nhập từ bàn phím cho đến khi giá trị nhập vào bằng 0 thì dừng lại và thông báo tổng tính được.

```

#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
int main(){
    ll x,S=0;
    do{
        system("cls"); //Lệnh xóa màn hình
        cout<<"\nNhap gia tri tu ban phim x="; cin>>x;
        S+=x;
    } while(x!=0);
    cout<<"\nTong cac so vua nhap la:"<<S;
return 0;}

```

Bài 22: Một người gửi tiết kiệm không kỳ hạn với số tiền G VNĐ với lãi suất 0.0083% mỗi tháng. Hỏi sau bao nhiêu tháng thì có thể rút số tiền lớn hơn hoặc bằng R VNĐ.

```

#include<bits/stdc++.h>
using namespace std;
int main(){
    int m=0;
    double G,R,T;
    cout<<"\nNhap so tien gui G="; cin>>G;
    do{

```

```

    cout<<"\nNhap so tien rut R="; cin>>R;
}while(R<=G);
T=G;
while(T<R){
    m++;
    T+=G*0.0083; //Lãi suất không kỳ hạn 0.1%/Năm
}
cout<<"\nSau "<<m<<" thang se rut duoc so tien theo Y/C!";
return 0;}

```

Bài 23: Phân tích số $N \in \mathbb{Z}^+$ thành tích các thừa số nguyên tố, N nhập từ bàn phím.

```

#include<bits/stdc++.h>
using namespace std;
int main(){
    int n=60; int i=2;
    while(i<=n){
        if (i<n){
            while (n%i==0){
                cout<<i<<"*";
                n/=i;}
            } else cout<<i;
            i++;
        }
    }
return 0;}

```

Bài 24: MENU tự tạo

```

#include<bits/stdc++.h>
using namespace std;
//Chương trình con dạng hàm tính diện tích hình vuông
float dienTichHV(){
    float a;
    cout<<"\n\tCHUONG TRINH TINH DIEN TICH HINH VUONG";
    cout<<"\nNhap canh a=";
    cin>>a;
    return a*a;
}
//Chương trình con dạng thủ tục tính diện tích hình tròn
void dienTichHT(){
    float s,r;
    cout<<"\n\tCHUONG TRINH TINH DIEN TICH HINH TRON";
    cout<<"\nNhap ban kinh R=";
    cin>>r;
    s=3.14*r*r;
    cout<<"\nDien tich hinh trong la:"<<setprecision(2)<<fixed<<s;
}
//CTC dạng thủ tục không có giá trị trả về, bắt đầu bằng void.
void taoMenu(){

```



```

    cout<<"\n\t\t1.Tinh dien tich hinh vuong!";
    cout<<"\n\t\t2.Tinh dien tich hinh tron!";
    cout<<"\n\t\t3.Tinh dien tich hinh CN!";
    cout<<"\n\t\t4.Tinh dien tich hinh Tam Giac!";
}
//Chương trình 2
void luaChon() {
    int chon;
    cout<<"\nMoi ban nhap lua chon:"; cin>>chon;
    switch(chon) {
        case 1: cout<<"Dien tich Hinh vuong la:"<<dienTichHV(); break;
        case 2: dienTichHT(); break;
        default: cout<<"Chương trình dang cap nhat chuc nang nay!";
    }
}
int main() {
    char tl;
    do{
        system("cls"); //Xóa màn hình
        cout<<"\t\t\tCHAO CAC BAN DEN VOI C++!\n"; //\t tab cách ra
        taoMenu(); //Hiện Menu các lựa chọn
        luaChon();
        cout<<"\nBAN CO MUON TIEP TUC KHONG [C/K]!";
        cin>>tl;
    }while(tl!='K' && tl!='k');
return 0;}

```

Bài 25: Tìm và in số hoàn hảo

Tìm và in ra màn hình tất cả các số hoàn hảo không vượt quá số **N** cho trước nhập từ bàn phím. Số hoàn hảo là số bằng tổng các ước số của nó không kể ước số là chính số đó. Ví dụ số 6 là số hoàn hảo vì $6=1+2+3$.

```

/* 6 28 496 8128 33550336 8589869056 137438691328 */
#include<bits/stdc++.h>
using namespace std;
typedef long long ll; // Định nghĩa ll kiểu long long
//#define ll long long // Cách khác định nghĩa ll kiểu long long
bool SoHoanHao(ll x) {
    ll s=0;
    for(ll j=1;j<x;j++)
        if(x%j==0) s+=j;
    if(s==x) return true;
    else return false;
}
int main() {
    ll n;
    cout<<"\nBan nhap vao mot so nguyen N="; cin>>n;
    for(ll i=1;i<=n;i++){
        if(SoHoanHao(i)) cout<<i<<" ";
    }
}

```

```
}  
return 0;}
```

PHẦN 3 - MẢNG MỘT CHIỀU

Lập chương trình nhập vào mảng một chiều A gồm N số nguyên:

Bài 0: Khai báo, nhập, xuất mảng 1 chiều A

```
#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int main() {
    int a[nMax]; //Khai báo mảng chứa tối đa 100 số nguyên
    int n = 0; //Số phần tử của mảng
    cout << "\nNhập số phần tử N=";
    cin >> n;
    //Nhập n phần tử cho M1C a
    for(int i=0;i<n;i++){
        cout<<"A["<<i+1<<"]="; //chỉ số 0 nhưng phần tử thứ 1 (i+1)
        cin>>a[i];
    }
    //Xuất n phần tử M1C a
    cout<<"\t\tCAC PHAN TU CUA M1C A LA:"; // \t\t đặt tab 2 lần
    for(int i=0;i<n;i++) cout<<a[i]<<" ";
    return 0;}

```

Bài 1: Tìm Max, Min của M1C A.

N=7	Min=2
5 8 9 2 17 11 20	Max=20

```
#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax];
void NhapM1C(int a[]){
    cout << "\nNhập số phần tử N=";
    cin >> n;
    for(int i=0;i<n;i++){
        cout<<"A["<<i+1<<"]=";
        cin>>a[i];
    }
}
int main() {
    NhapM1C(a);
    //Tìm giá trị lớn nhất, nhỏ nhất
    int vmax,vmin,vt1,vt2;
    vmax=a[0]; vt1=0;
    vmin=a[0]; vt2=0;
    for(int i=1;i<n;i++){
        if(vmax<a[i]) {vmax=a[i];vt1=i;}
        if(vmin>a[i]) {vmin=a[i];vt2=i;}
    }
    cout<<"GTLN Max="<<vmax<<" Tai vi tri:"<<vt1+1;
    cout<<"\nGTNN Min="<<vmin<<" Tai vi tri:"<<vt2+1;
    return 0;}

```

Bài 2: Tìm UCLN của M1C A.

N=3 4 8 12	UCLN mang la:4
---------------	----------------

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax];
void NhapM1C(int a[]){
    do{
        system("cls");
        cout << "\nNhap so phan tu >=2,N="; cin >> n;
    }while(n<2); //Phải ít nhất 2 phần tử!

    for(int i=0;i<n;i++){
        cout<<"A["<<i+1<<"]="";
        cin>>a[i];
    }
}
int UCLN(int x,int y){
    while(x!=y)
        if(x>y) x-=y; else y-=x;
return x;}
int main() {
    NhapM1C(a);
    int d=UCLN(a[0],a[1]);
    for(int i=2;i<n;i++) d=UCLN(d,a[i]);
    cout<<"UCLN cua day la:"<<d;
return 0;}

```

Bài 3: Sắp xếp M1C A thành dãy không giảm (Nổi bọt, tráo đổi).

N=7 5 8 9 2 17 11 20	Day da sap xep: 2 5 8 9 11 17 20
-------------------------	-------------------------------------

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax];
void NhapM1C(int x[]){
    cout << "\nNhap so phan tu N="; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
}
void XuatM1C(int x[]){
    cout<<"\t\tCAC PHAN TU CUA M1C X LA:";
    for(int i=0;i<n;i++) cout<<x[i]<<" ";
}

```

```

int main() {
    NhapM1C(a);
    //Thuật toán sắp xếp
    for(int i=0;i<n-1;i++)
        for(int j=i+1;j<n;j++)
            if(a[i]>a[j]){
                int tam=a[i];
                a[i]=a[j];
                a[j]=tam;
            }
    XuatM1C(a);
return 0;}

```

Bài 4: M1C A đã sắp xếp không giảm, nhập số nguyên K. Chèn K vào M1C A sao cho M1C A vẫn giữ được tính chất không giảm.

N=7; K=15	Day sau khi chen la:
5 8 9 2 17 11 20	2 5 8 9 11 15 17 20

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax];
void NhapM1C(int x[]){
    cout << "\nNhap so phan tu N="; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
}
void SapXep(int x[],int n){
    //Thuật toán sắp xếp
    for(int i=0;i<n-1;i++)
        for(int j=i+1;j<n;j++)
            if(a[i]>a[j]){
                int tam=a[i];
                a[i]=a[j];
                a[j]=tam;
            }
}
void XuatM1C(int x[],int n){
    cout<<"\t\tCAC PHAN TU CUA M1C LA:";
    for(int i=0;i<n;i++) cout<<x[i]<<" ";
}
int main() {
    NhapM1C(a);
    SapXep(a,n);
    XuatM1C(a,n);
    //Chèn phần tử vào mảng đã sắp xếp
    int k,r,vt=0;
    cout<<"\nNhap so can chen K="; cin>>k;
    while(k>a[vt] & vt<n) vt++; //Tìm vị trí cần chèn
    if (vt==n) a[n]=k;
    else {

```

```

        r=n; //Tăng lên 1 phần tử
        while(r>vt){
            a[r]=a[r-1];
            r--;
        }
        //Chèn K vào vị trí i
        a[vt]=k;
    }
    XuatM1C(a,n+1);
return 0;

```

Bài 5: Nhập số nguyên K. Tìm xem K có xuất hiện trong dãy A không, nếu có thông báo vị trí i sao cho $A_i = K$.

N=7; K=9 5 8 9 2 17 11 20	9 xuất hiện tại vị trí 3
N=7; K=1 5 8 9 2 17 11 20	1 không xuất hiện trong mảng

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax];
void NhapM1C(int x[]){
    cout << "\nNhập số phần tử N="; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]=";
        cin>>x[i];
    }
}
int main() {
    NhapM1C(a);
    //Nhập số nguyên K
    int k,pos=-1;
    cout<<"Nhập số nguyên K="; cin>>k;
    for(int i=0;i<n;i++)
        if(a[i]==k){
            pos=i;break;} //Tìm thấy thoát khỏi vòng lặp
    if(pos==-1) cout<<k<<" không xuất hiện trong mảng!";
    else cout<<k<<" xuất hiện tại vị trí "<<pos+1;
return 0;
}

```

Bài 6: Tìm phần tử xuất hiện nhiều lần nhất trong M1C A.

N=9 2 5 3 2 5 11 2 21 2	2 xuất hiện 4 lần
----------------------------	-------------------

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax],d[nMax];
bool mark[nMax];
void NhapM1C(int x[]){
    cout << "\nNhập số phần tử N="; cin >> n;

```

```

        for(int i=0;i<n;i++){
            cout<<"X["<<i+1<<"]="";
            cin>>x[i];
        }
    }
int main() {
    NhapM1C(a);
    memset(mark,false,sizeof(mark));
    memset(d,0,sizeof(d));
    for(int i=0;i<n-1;i++)
        for(int j=i+1;j<n;j++){
            if (mark[j]==false && a[i]==a[j]) {
                d[i]++;
                mark[j]=true;
            }
        }
    //Tìm đồng thời số lượng max, số lượng min để xét trường hợp xuất hiện như nhau
    int slmax=d[0],vt=0,slmin=d[0];
    for(int i=1;i<n;i++){
        if(d[i]>0) {
            if(slmax<d[i]){slmax=d[i]; vt=i;}
            if(slmin>d[i]){slmin=d[i];}
        }
    }
    if(slmax==slmin) cout<<"Cac phan tu xuat hien nhu nhau!";
    else
        cout<<"Phan tu xuat hien Max la:"<<a[vt]<<" so luong la:"<<slmax+1;
return 0;}

```

Bài 7: Tính tổng các phần tử chia hết cho k (k nhập từ bàn phím).

N=7; K=3 5 8 9 2 3 11 21	Tong cac so chia het 3 la: 33
-----------------------------	-------------------------------

*** Gợi ý:**

- Nhập M1C
- Với mỗi a[i], nếu a[i] chia hết k thì s+=a[i];

Bài 8: Tìm phần tử lớn nhì trong M1C A.

N=7 5 8 9 2 17 11 20	Max2=17
-------------------------	---------

*** Gợi ý:**

- Nhập M1C
 - Tìm phần tử Max
 - Tìm phần tử Max2 khác Max
- (Sắp xếp không giảm M1C, tìm vị trí đầu tiên mà a[i]!=Max, nếu vị trí>=0 thì có phần tử Max2, ngược lại không có)

Bài 9: Hiện ra màn hình các phần tử là số nguyên tố trong M1C A.

N=7 5 8 9 2 17 11 20	Cac SNT la: 5 2 17 11
-------------------------	-----------------------

*** Gợi ý:**

- Nhập M1C
- Duyệt M1C, nếu a[i] là SNT thì << a[i]

Bài 10: Hiện ra màn hình các số chính phương trong M1C A.

N=7 5 8 9 2 1 11 16	Cac SCP la: 9 1 16
------------------------	--------------------

*** Gợi ý:**

- Nhập M1C
- Duyệt M1C, nếu a[i] là SCP thì << a[i]

Bài 11: Kiểm tra dãy A có là cấp số cộng (CSC) hay không, nếu có đưa ra công sai (d).

N=5 2 5 8 11 14	Day la CSC voi d=3
N=7 5 8 9 2 1 11 16	Khong la CSC

```
#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax];
void NhapM1C(int x[]){
    do{
        system("cls");
        cout << "\nNhap so phan tu N>=2: "; cin >> n;
    } while(n<2);
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
}
int main() {
    NhapM1C(a);
    int d=a[1]-a[0],i=2; //Tinh công sai
    while(a[i]-a[i-1]==d && i<n) i++;
    if(i==n) cout<<"Day la CSC!";
    else cout<<"Day khong la CSC!";
return 0;}
```

Bài 12: Trộn hai mảng đã được sắp xếp theo thứ tự tăng dần lại thành một mảng mới mà vẫn đảm bảo thứ tự của nó. (Chú ý: không dùng thuật toán sắp xếp)

N=3; A: 2 5 8	Mang sau khi tron la:
N=4; B: 1 3 9 15	1 2 3 5 8 9 15

```
#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
#define nmMax 200
int n,m,a[nMax],b[nMax],c[nmMax];
void NhapM1C(int x[],int y[]){
    cout << "\nNhap N phan tu va M1C tang dan:"; cin >> n;
```



```

    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
    cout << "\nNhap Mphan tu va M1C tang dan:"; cin >> m;
    for(int i=0;i<m;i++){
        cout<<"Y["<<i+1<<"]="";
        cin>>y[i];
    }
}
void XuatM1C(int x[],int nm){
    cout<<"\t\tKET QUA SAU KHI GOP:";
    for(int i=0;i<nm;i++) cout<<x[i]<<" ";
}
int main() {
    NhapM1C(a,b);
    int l=0,r=0,k=0;
    while(l<n || r<m){
        if(l==n & r<m) {c[k]=b[r];r++;k++;}
        else if(l<n & r==m) {c[k]=b[l];l++;k++;}
        else
            if(a[l]<=b[r]) {c[k]=a[l];l++;k++;}
            else {c[k]=b[r]; r++;k++;}
    }
    XuatM1C(c,n+m);
return 0;}

```

Bài 13: Tìm dãy số nguyên dương liên tiếp, không giảm, dài nhất trong mảng, ghi ra chỉ số đầu tiên và cuối cùng của dãy tìm được.

N=9 1 1 4 5 4 2 3 4 5	L=1 R=4
N=20 7 9 10 3 4 4 7 8 9 10 8 12 12 15 17 19 10 20 11 22	L=4 R=10

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int L,R,n,a[nMax];
void NhapM1C(int x[]){
    cout << "\nNhap Nphan tu va M1C:"; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
    L=0; R=n;
}
void Xuli(int x[],int &l,int &r){
    int d=0,dmax=0;
    for(int i=0;i<n-1;i++){
        if(a[i+1]>=a[i]) d++;
        else if (dmax<d) {

```

```

        dmax=d;
        r=i+1;
        l=i-dmax+1;
        d=0;
    }
}
}
int main() {
    NhapM1C(a);
    Xuli(a,L,R);
    cout<<L<<"--"<<R;
return 0;}

```

Bài 14: Tìm số phần tử nhỏ hơn phần tử tại vị trí thứ i trong mảng

N=4; A:110 120 130 140	0 1 2 3
	Vị trí một không có phần tử nào nhỏ hơn nó
	Vị trí hai có 1 phần tử nhỏ hơn nó
	Vị trí ba có 2 phần tử nhỏ hơn nó
	Vị trí 4 có 3 phần tử nhỏ hơn nó

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
int n,a[nMax],d[nMax];
void NhapM1C(int x[]){
    cout << "\nNhap Nphan tu va M1C:"; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
}
void Xuli(int x[]){
    memset(d,0,sizeof(d)); //Khởi tạo tất cả các phần tử đều là 0
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            if (a[i]>a[j]) d[i]++;
        }
        for(int i=0;i<n;i++) cout<<d[i]<<" ";
    }
int main() {
    NhapM1C(a);
    Xuli(a);
return 0;}

```

Bài 15: Tìm tổng lớn nhất của K phần tử liên tiếp trong dãy.

N=5; K=2	8
A: 4 3 2 6 1	

```

#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
#define ll long long
int n,k,vt,a[nMax];

```

```

ll smax=0;
void NhapM1C(int x[]){
    cout << "\nNhap Nphan tu va M1C:"; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];
    }
    cout<<"Nhap K="; cin>>k;
}
void Xuli(int x[],int k){
    for(int i=0;i<n;i++){
        ll s=0;
        for (int j=0;j<k;j++)
            s+=a[i+j];
        if(smax<s) {
            smax=s;
            vt=i;
        }
    }
    cout<<"Smax="<<smax<<" Tu:"<<vt+1<<"-->"<<vt+k;
}
int main() {
    NhapM1C(a);
    Xuli(a,k);
    return 0;}

```

Bài 16: Tìm số Fibonacci lớn nhất trong một dãy gồm N số nguyên dương.

N=9
A: 2 3 4 6 9 8 12 55 10
(Note: A[i]≤1018)
(fibo 87 excel ~1018)

55

```

Max int=2147483647
Min int=-2147483648
Max long long=9223372036854775807
Min long long=-9223372036854775808
Max foat=3.40282e+038
Min foat=1.17549e-038
Max double=1.79769e+308
Min double=2.22507e-308

```

```

/*
N=9; A: 2 3 4 6 9 8 12 55 10
- Tim Max(A)
- Tao ra M1C cac so fibonaxi nho hon bang max (A)
- Tim a[i] la so fibo max
*/
#include<bits/stdc++.h>
using namespace std;
#define nMax 100 //Khai báo hằng nMax = 100
#define ll long long
int n;
ll a[nMax],fib[nMax],fx;
ll myMax(ll x[]){
    ll amax=0;
    for(int i=0;i<n;i++){
        if(amax<a[i]) amax=a[i];}
    return amax;}
void NhapM1C(ll x[]){

```

```
    cout << "\nNhap N phan tu va M1C:"; cin >> n;
    for(int i=0;i<n;i++){
        cout<<"X["<<i+1<<"]="";
        cin>>x[i];}
    fx=myMax(x);
}
void Xuli(ll x[]){
    int m=1;
    ll fo=0;
    //Tạo mảng số fibo nhỏ hơn = max
    memset(fib,0,sizeof(fib));
    fib[0]=1; fib[1]=1;
    while(fib[m]<=fx){
        m++;
        fib[m]=fib[m-1]+fib[m-2];
    }
    //Tìm số fibo max
    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)
            if(a[i]==fib[j] && fib[j]>fo) fo=a[i];
    cout<<"Fibonaxi Max="<<fo;
}
int main() {
    NhapM1C(a);
    Xuli(a);
return 0;}
```

PHẦN 4 - KIỂU XÂU (~10⁸ kí tự)

Lập chương trình nhập vào một chuỗi kí tự bất kỳ:

Bài 1: Kiểm tra chuỗi đó có đối xứng gương hay không.

```

/*
 st.length()/st.size()-> trả về độ dài chuỗi
 getline(cin,st); -> nhập toàn bộ chuỗi.
 (Nếu nhập bằng cin, chỉ lấy được từ đầu chuỗi đến khoảng trắng đầu tiên.
 */
#include<bits/stdc++.h>
using namespace std;
string s="",st;
int main(){
    cout<<"Nhập vào một chuỗi:";
    getline(cin,st); //Nếu dùng cin thì chỉ lấy được từ đầu đến dấu cách
    for(int i=0;i<st.length();i++) s=st[i]+s;
    cout<<"\nChuỗi đảo ngược là:"<<s;
return 0;}

```

Bài 2: Đếm số kí tự là chữ số trong chuỗi.

```

/*
 n.find(st[i]) -> st[i] có là các số không (vị trí 0-9)
 */
#include<bits/stdc++.h>
using namespace std;
string st,n="0123456789";
int main(){
    int d=0;
    cout<<"Nhập vào một chuỗi:";
    getline(cin,st);
    for(int i=0;i<st.length();i++)
        //if(st[i]>='0' && st[i]<='9') d++;
        //if(n.find(st[i])<10) d++; //
        if(st[i]>=48 && st[i]<=57) d++;
    cout<<"\nSố chữ số trong chuỗi là:"<<d;
return 0;}

```

Bài 3: Đếm số kí tự là nguyên âm trong chuỗi. (Các chữ cái là nguyên âm: UEOAI)

```

/*
 na.find(st[i]) -> st[i] có là các nguyên âm
 */
#include<bits/stdc++.h>
using namespace std;
string st,na="ueoaiEUOAI";
int main(){
    int d=0;
    cout<<"Nhập vào một chuỗi:";
    getline(cin,st);
    for(int i=0;i<st.length();i++)
        if(na.find(st[i])<10) d++;
    cout<<"\nSố nguyên âm là:"<<d;
return 0;}

```

Bài 4: Hiện ra màn hình xâu đó dạng chữ in hoa & in thường.

```

/*
transform(s.begin(), s.end(),s.begin(), ::toupper); //Thu tục doi xau s sang chu HOA
transform(s.begin(), s.end(),s.begin(), ::tolower); //Thu tục doi xau s sang chu thuong
*/
#include<bits/stdc++.h>
using namespace std;
string st,shoa,sth;
int main(){
    cout<<"Nhap vao mot xau:";
    getline(cin,st);
    shoa=st; sth=st;
    //C1: Thuật toán đổi
    for(int i=0;i<st.length();i++){
        if(shoa[i]>='a' && shoa[i]<='z') shoa[i]=char(shoa[i]-32);
        if(sth[i]>='A' && sth[i]<='Z') sth[i]=char(sth[i]+32);
    }
    //C2: Sử dụng hàm có sẵn
    /*
transform(shoa.begin(),shoa.end(),shoa.begin(),::toupper);
transform(sth.begin(),sth.end(),sth.begin(),::tolower);
*/
    cout<<"\nXau in HOA la:"<<shoa;
    cout<<"\nXau in thuong la:"<<sth;
return 0;}

```

Bài 5: Hiện ra màn hình mỗi từ trên một dòng.

```

/*
st.find(' ',vt) -> Hàm trả về vị trí đầu tiên ' ' xuất hiện trong st từ vt
st.substr(vt,n) -> Hàm trả về xâu con của st, từ vị trí vs lấy n kí tự
*/
#include<bits/stdc++.h>
using namespace std;
string st;
int main(){
    int vt=0,vs=0;
    cout<<"Nhap vao mot xau:";
    getline(cin,st);
    vt=st.find(' ',vt);
    cout<<"\n"<<st.substr(vs,vt); //Từ đầu tiên
    while(vt<st.length()) {
        vs=vt+1;
        vt=st.find(' ',vs);
        if (vt>=0) cout<<"\n"<<st.substr(vs,vt-vs+1);
        else cout<<"\n"<<st.substr(vs,st.length()-vs+1); //Từ cuối cùng
    }
return 0;}

```

Bài 6: Tính tổng tất cả các chữ số trong xâu.

```

#include<bits/stdc++.h>
using namespace std;
string st,so="0123456789";
#define ll long long
int main(){

```

```

    ll s=0;
    cout<<"Nhap vao mot xau chuan:";
    getline(cin,st);
    for(int i=0;i<st.length();i++)
        if(so.find(st[i])<10) s+=st[i]-48; //trừ 48 ra số tương ứng
    cout<<"Tong cac chu so la:"<<s;
return 0;

```

Bài 7: Đếm số từ có trong xâu.

Xem bài 5

Bài 8: Chuẩn hóa lại xâu vừa nhập.

|st = ' _CHUC_muNg_nAm_Moi_' → St= 'Chuc_Mung_Nam_Moi'

```

#include<bits/stdc++.h>
using namespace std;
string s;
void KhoiTao() {
    freopen("DATA.INP","r",stdin); //Mo file DATA.INP de doc du lieu
    freopen("DATA.OUT","w",stdout); //Mo file DATA.OUT de ghi du lieu
    getline(cin,s); // Doc xau ki tu tu file vao bien st
}
void XoaKhoangTrang(string &st) {
    string bk=" ";
    while(st.find(bk,0)<st.length()){// Tim thay 2 blank thi tra ve gia tri <length
        st.erase(st.begin()+st.find(bk,0));
    }
    if(st[0]==' ') st.erase(st.begin()+0);
    if(st[st.length()-1]==' ') st.erase(st.begin()+st.length()-1);
}
void DoiThuongHoaDauTu(string &st) {
    for(int i=0;i<=st.length();i++)
        if(st[i]>='A' && st[i]<='Z') st[i]+=32; //Kt HOA+32 = Kt thuong
    // Doi hoa dau tu
    if(st[0]>='a' && st[0]<='z') st[0]-=32; //Kt thuong-32 = Kt HOA
    for(int i=2;i<st.length();i++)
        if(st[i+1]>='a' && st[i+1]<='z' && st[i]==' ') st[i+1]-=32;
}
int main() {
    KhoiTao();
    XoaKhoangTrang(s);
    //transform(s.begin(), s.end(),s.begin(), ::toupper); //Thu tuc doi xau s sang chu HOA
    //transform(s.begin(), s.end(),s.begin(), ::tolower); //Thu tuc doi xau s sang chu thuong
    DoiThuongHoaDauTu(s);
    cout<<s;
return 0;
}

```

Bài 9: Xâu nhập vào có dạng dãy bit (0,1) Đưa bit 1 về đầu xâu, bit 0 về cuối xâu.

Vd: St= '1010010' → St='1110000'

```

/*
Dua bit '1' ve dau, bit '0' ve cuoi
st="1001010" -> st="1110000"

```

```

*/
#include<bits/stdc++.h>
using namespace std;
string st,s="";
#define ll long long
bool Not_Bit(string s){
    for(int i=0;i<s.length();i++)
        if(s[i]!='0' && s[i]!='1') return true; //Không là dãy bit
}
int main(){
    do{
        system("cls");
        cout<<"Nhap vao day bit:";
        getline(cin,st);
    }while(Not_Bit(st));
    for(int i=0;i<st.length();i++)
        if(st[i]=='0') s=s+'0';
        else s='1'+s;
    cout<<"Xau ket qua la:"<<s;
return 0;}

```

Bài 10: Cho biết kí tự xuất hiện nhiều lần nhất trong xâu.

Tương tự phần tử xuất hiện nhiều lần nhất trong M1C (Bài 6)

Bài 11: Nhập xâu cần tìm s1, nhập xâu cần thay thế s2. Tìm và thay thế tất cả các xâu s1 bằng xâu s2.

```

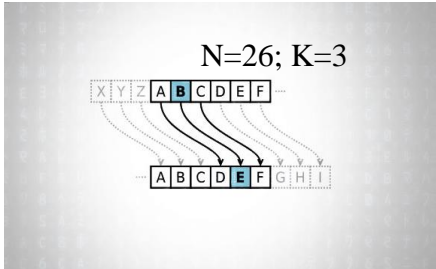
/*
size_t -> Kiểu dữ liệu không dấu linh hoạt tu động mô rộng khi cần
Tìm kiếm và thay thế xâu s1 có trong st bằng xâu s2!
st.replace(vt,n,s) -> Thay thế xâu s trong st tại vị trí vt n kí tự
string::npos -> Hàng số trả về khi hàm find(không tìm thấy)
*/
#include<bits/stdc++.h>
using namespace std;
string st,s1,s2;
int main(){
    cout<<"Nhap mot xau bat ky:";
    getline(cin,st);
    do{
        cout<<"\nNhap xau can tim:";
        getline(cin,s1);
    } while(st.find(s1,0)>st.length());
    cout<<"Nhap xau can thay the:";
    getline(cin,s2);
    size_t pos=0,oo=string::npos;
    while((pos=st.find(s1,pos))!=oo){ //xau s1 co xuat hien trong st //
<st.length() cung OK
        st.replace(pos,s1.length(),s2); //thay the xau s1 bang xau s2 trong st
        pos+=s2.length(); //vi tri tim ke tiep trong xau st
    }
    cout<<"Xau sau khi thay the la:"<<st;
return 0;}

```


Bài 12: Nhập vào khóa K, hãy mã hóa xâu vừa nhập theo phương pháp mã hóa Ceda:

Phương pháp mã hóa Ceda: (Áp dụng cho bảng gồm 26 chữ cái tiếng Anh)

- Thay thế mỗi chữ cái bằng chữ cái đứng sau nó **k** vị trí trong bảng chữ cái. Việc tìm kiếm và thay thế tiến hành vòng tròn theo bảng chữ cái.
- Các ký tự ngoài 26 chữ cái tiếng Anh được giữ nguyên (*không thay thế*)



Ví dụ: Với N=26; K=3

Văn bản gốc: BAD 09-11

Văn bản mã: EDG 09-11

```
#include<bits/stdc++.h>
using namespace std;
char tt; int K;
string s,ttk,kk;
void khaiTao(){
    freopen("CEDA.INP","r",stdin); //Mo file CEDA.INP de doc du lieu
    freopen("CEDA.OUT","w",stdout); //Mo file CEDA.OUT de ghi du lieu
    getline(cin,ttk); //Doc dong dau tien trong tep
    tt=ttk[0]; //Trang thai tt = ki tu dau tien
    //ttk.substr(vt,n) --> copy xau ttk tu vi tri vt, n ki tu; stoi -->chuyen xau sang int
    K=stoi(ttk.substr(2,ttk.length()-2))%26;
    getline(cin,s); }
string enCodeCeda(string st,int k){
    string est="";
    transform(st.begin(), st.end(),st.begin(), ::tolower);
    for(int i=0;i<st.length();i++){
        if (st[i]>='a' && st[i]<='z')
            //Ma Ascii cua 'z' =122, st[i]+k>122 thi vong lai tu a...
            est+=char(st[i]+k>122 ? (st[i]+k) % 122 + 96:st[i]+k);
        else est+=st[i]; }
    return est; }
string deCodeCeda(string st,int k){
    string dst="";
    for(int i=0;i<st.length();i++){
        if (st[i]>='a' && st[i]<='z')
            //Ma Ascii cua 'z' =122, st[i]+k>122 thi vong lai tu a...
            dst+=char(st[i]-96-k >0 ? st[i]-k : st[i]-96-k+122);
        else dst+=st[i]; }
    transform(dst.begin(), dst.end(),dst.begin(), ::toupper);
    return dst; }
void edCodeCeda(string st,char tt,int k){
    if(tt=='E') cout<<enCodeCeda(st,k);
    if(tt=='D') cout<<deCodeCeda(st,k);
}
```

```
int main() {
    khaiTao();
    edCodeCeda(s,tt,K);
return 0;}

```

Bài 13: Nhập vào một năm dương lịch N, đổi sang năm âm lịch tương ứng.

```
/*
 size_t -> Kieu du lieu khong dau linh hoat tu dong mo rong khi can
*/
#include<bits/stdc++.h>
using namespace std;
string can[10]={"Quy","Giap","At","Binh","Dinh","Mau","Ky","Canh","Tan","Nham"};
string chi[12]={"Hoi","Ti","Suu","Dan","Mao","Thin","Ty","Ngo","Mui","Than","Dau","Tuat"};
int main() {
    size_t n;
    string al="";
    cout<<"Nhap mot xau bat ky:"; cin>>n;
    al=can[(n-3)%10]+ " "+chi[(n-3)%12];
    cout<<"Nam "<<n<<" la nam "<<al<<"!";
return 0;}

```

Bài 14: Nhập vào một xâu chứa các chữ số, kiểm tra xem xâu vừa nhập có là SỐ không?

```
#include<bits/stdc++.h>
using namespace std;
string st;
int main() {
    cout<<"Nhap vao mot so:";
    getline(cin,st);
    bool oknum=true;
    for(int i=0;i<st.length();i++)
        if (st[i]<'0' || st[i]>'9'){
            oknum=false;
            break;
        }
    if (oknum) cout<<"Xau vua nhap la 1 so!";
    else cout<<"Xua vua nhap khong la so!";
return 0;}

```

Bài 15: Nhập vào một số nguyên dương N, đọc số đó ($0 < N \leq 10^{11}$)

```
/*
 DOC SO
 hty = n / 1000000000
 htrieu = (n - hty * 1000000000) / 1000000
 htram = n % 1000
 nghin = ((n - htram) / 1000) % 1000
*/
#include<bits/stdc++.h>
using namespace std;
string st;
string num[]={ "", "mot", "hai", "ba", "bon", "nam", "sau", "bay", "tam", "chin"};

```

```

typedef long long ll;
ll n,dv,ch,tr,htram,hnghin,htrieu,hty;
string doc3so(ll so) {
    string kqdoc="";
    dv=so%10;
    ch=(so%100)/10;
    tr=so/100;
    if(tr>0) kqdoc+=num[tr]+" tram";
    switch (ch){
        case 0: if(tr>0 && dv>0) kqdoc+=" linh "; break;
        case 1: kqdoc+=" muoif "; break;
        default: kqdoc+=num[ch]+" muoi ";
    }
    switch (dv){
        case 0: break;
        case 1: if(ch<=1) kqdoc+=" mot"; else kqdoc+=" mots"; break;
        case 5: if(ch==0) kqdoc+=" nam"; else kqdoc+=" lam"; break;
        default: kqdoc+=num[dv];
    }
return kqdoc;}
//Ham doc so
string DocSo(ll so) {
    string kqdoc="";
    hty=so/1000000000;
    htrieu=(so-hty*1000000000)/1000000;
    htram= so % 1000;
    hnghin=((so-htram)/1000)%1000;
    if(hty>0) kqdoc+=doc3so(hty)+" ty ";
    if(htrieu>0) kqdoc+=doc3so(htrieu)+" trieu "; // xử lí tạo nữa làm biếng ☺
    if(hnghin>0) kqdoc+=doc3so(hnghin)+" nghin ";
    if(htram/100>0) kqdoc+=doc3so(htram);
    else
    { if(htram/100==0 && hnghin>0) kqdoc+="khong tram";
      if(htram/10>0) kqdoc+=doc3so(htram);
      else kqdoc+= " linh " + doc3so(htram);}
return kqdoc;
}
int main() {
    cout<<"Nhap vao mot so nguyen duong:"; cin>>n;
    cout<<n<<" Doc la: "<<DocSo(n);
return 0;}

```

Bài 16: Nhập vào 2 số nguyên lớn (tối đa 500 chữ số). Tính Tổng hai số vừa nhập.

```
/*
Chương trình cộng hai số lớn
//n1=stoll(so1); > Chuyển so1 dạng xâu sang số nguyên long ~18 chữ số!
*/
#include<bits/stdc++.h>
using namespace std;
typedef string myNum;
myNum so1,so2;
ll n1,n2;
bool isNumber(myNum so){
    for(int i=0;i<so.length();i++)
        if (so[i]<'0' || so[i]>'9')
            return false;
    return true;
}
void Nhap2So(){
    do{
        cout<<"\nNhập vào số nguyên thứ 1:"; getline(cin,so1);
    } while (! isNumber(so1));
    do{
        cout<<"\nNhập vào số nguyên thứ 2:"; getline(cin,so2);
    } while (! isNumber(so2));
    while(so1.length()<so2.length()) so1='0'+so1;
    while(so1.length()>so2.length()) so2='0'+so2;
}
//Hàm tính tổng hai số
string SumBigNum(myNum a, myNum b){
    myNum kq="";
    int t,nho=0;
    for(int i=a.length()-1;i>=0;i--){
        t=(a[i]+b[i]+nho-2*48)%10; //Đổi ra số - 2*48
        kq= char(t+48)+kq;
        nho=(a[i]+b[i]+nho-2*48)/10;
    }
    if(nho==1) kq='1'+kq;
    return kq;
}
int main(){
    Nhap2So();
    cout<<SumBigNum(so1,so2);
    return 0;}

```

Bài 17: Nhập vào 2 số nguyên lớn (tối đa 500 chữ số). Tính Hiệu hai số nguyên vừa nhập.

```
/*
  Chương trình trừ hai số lớn
  //n1=stoll(so1); > Chuyển so1 dạng xấu sang số nguyên long ~18 chữ số!
*/
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef string myNum;
myNum so1,so2,tam;
// ll n1,n2;
bool isNumber(myNum so) {
    for(int i=0;i<so.length();i++)
        if (so[i]<'0' || so[i]>'9')
            return false;
    return true;
}
void Nhap2So() {
    do{
        cout<<"\nNhập vào số nguyên thứ 1:"; getline(cin,so1);
    } while (! isNumber(so1));
    do{
        cout<<"\nNhập vào số nguyên thứ 2:"; getline(cin,so2);
    } while (! isNumber(so2));
    while(so1.length()<so2.length()) so1='0'+so1;
    while(so1.length()>so2.length()) so2='0'+so2;
    //Đảo vai trò so1>so2
    if(so1<so2) {
        tam=so1;
        so1=so2;
        so2=tam;
    }
}
//Hàm hiệu hai số lớn trừ số bé
string subtrBigNum(myNum a, myNum b) {
    myNum kq="";
    int t,vay=0;
    for(int i=a.length()-1;i>=0;i--){
        t=(a[i]-b[i]-vay); //Trừ bình thường
        if(t<0){ //số bị trừ nhỏ hơn số trừ phải mượn 1 chục
            t+=10;
            vay=1; //mượn là 1
        }
        else vay=0;
        kq= char(t+48)+kq;
    }
}
```

```

    while(kq[0]=='0' && kq.length()>1) kq.erase(kq.begin()+0); //Xóa số 0 ở đầu
    // if(kq=="") kq='0'; //Kết quả rỗng thì gán kq=0
    return kq;
}
int main(){
    Nhap2So();
    //n1=stoll(so1); n2=stoll(so2);
    //cout<<n1-n2<<"\n";
    cout<<subtrBigNum(so1,so2);
return 0;}

```

Bài 18: Nhập vào 2 số nguyên lớn (tối đa 500 chữ số). Tính Tích hai số nguyên vừa nhập.

```

/*
Chương trình NHAN hai so lon
- Hàm cộng 2 số lớn: SumBigNum(x,y).
- Hàm nhân 1 số lớn với 1 số nhỏ: MulBigx(x,n).
- Khởi tạo: nt= Mul2BigNum(n1,n2[len-1]).
* Lặp: k=len-2-->0
- o+='0' ->Thêm kí tự '0'
- tính ns=Nhân số (n1,n2[k])+o;
- tích=SumBigNum(nt,ns);
- nt=tích -> Cập nhật lại nt
*/
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef string myNum;
myNum so1,so2;
ll n1,n2;
bool isNumber(myNum so){
    for(int i=0;i<so.length();i++)
        if (so[i]<'0' || so[i]>'9')
            return false;
    return true;
}
void Nhap2So(){
    do{
        cout<<"\nNhap vao so nguyen thu 1:"; getline(cin,so1);
    } while (! isNumber(so1));
    do{
        cout<<"\nNhap vao so nguyen thu 2:"; getline(cin,so2);
    } while (! isNumber(so2));
}
//Hàm tính tổng hai số
myNum SumBigNum(myNum a, myNum b){
    myNum kq="";
    int t,nho=0;
    while(a.length()<b.length()) a='0'+a;

```

```

while(a.length()>b.length()) b='0'+b;
for(int i=a.length()-1;i>=0;i--){
    t=(a[i]+b[i]+nho-2*48)%10; //Đổi ra số - 2*48
    kq= char(t+48)+kq;
    nho=(a[i]+b[i]+nho-2*48)/10;
}
if(nho==1) kq='1'+kq;
return kq;
}
//Ham nhan 1 so lon voi 1 so nho
myNum MulBigx(myNum x, int n){
    myNum c="";
    int t1,nho1=0;
    for(int j=x.length()-1;j>=0;j--){
        t1=((x[j]-48)*n+nho1)%10;
        c=char(t1+48)+c;
        nho1=((x[j]-48)*n+nho1)/10;
    }
    if(nho1>0) c=char(nho1+48)+c;
    if(c=="") for(int j=0;j<x.length();j++) c+='0';
    return c;
}
myNum Mul2BigNum(myNum n1,myNum n2){
    myNum o="",nt,ns,tich="";
    nt=MulBigx(n1,(n2[n2.length()-1]-48));
    cout<<nt;
    for(int k=n2.length()-2;k>=0;k--){
        o+='0';
        ns=MulBigx(n1,(n2[k]-48))+o; //Ghép số 0 vào số mới sau mỗi lần nhân
        cout<<"\n"<<ns;
        tich=SumBigNum(nt,ns);
        nt=tich;
    }
    if(tich=="") tich="0";
    return tich;
}
int main(){
    Nhap2So();
    cout<<"\n"<<Mul2BigNum(so1,so2);
return 0;}

```

Để xử lý số lớn chúng ta sẽ chuyển sang xử lý chuỗi ký tự.

Trong code dưới đây các bạn chú ý phương thức **str.insert(int pos, int n, int ch)**; chèn n lần ký tự ch vào vị trí pos của chuỗi str;

Ngoài ra còn có 2 phương thức khác là :

str.insert(int pos, char* s); chèn s (mảng ký tự kết thúc "\0") vào vị trí pos của str;

str.insert(int pos, string s); chèn chuỗi s (kiểu string) vào vị trí pos của chuỗi str;

```
01 #include <string>
02 #include <iostream>
03 using namespace std;
04
05 int stringToNum(char c) // chuyen char sang so
06 {
07     return c - '0';
08 }
09
10 char numToString(int n) // chuyen so sang char
11 {
12     return (char)(n+48);
13 }
14
15 void chuanHoa(string &a, string &b) // lam 2 xau co do dai bang nhau
16 {
17     int l1 = a.length(), l2 = b.length();
18     if (l1 >= l2)
19     {
20         b.insert(0, l1-l2, '0'); // chen vao dau cua b cac ky tu '0'
21     }
22     else
23     {
24         a.insert(0, l2-l1, '0'); // chen vao dau cua a cac ky tu '0'
25     }
26 }
27
28 string sum(string a, string b) // tong 2 so
29 {
30     string s = "";
31     chuanHoa(a,b); // chuan hoa
32     int l = a.length();
33
34     int temp = 0;
35     for (int i=l-1; i>=0; i--) // duyet va cong
36     {
37         temp = stringToNum(a[i]) + stringToNum(b[i]) + temp; // tinh tong tung doi
38         s.insert(0,1,numToString(temp%10)); // gan phan don vi vao
39         temp = temp/10; // lay lai phan hang chuc
40     }
41     if (temp>0) // neu hang chuc > 0 thi them vao KQ
42     {
43         s.insert(0,1,numToString(temp));
44     }
45     return s;
46 }
47
48 // nhan so co 1 chu so voi so co nhieu chu so (VD 4 va 7826), lam tuong tu nhu phep c
49 string nhanNho(char a, string b)
50 {
51     string s = "";
52     int temp = 0;
53     for (int i=b.length()-1; i>=0; i--)
54     {
55         temp = stringToNum(a) * stringToNum(b[i]) + temp;
56         s.insert(0,1,numToString(temp%10));
57         temp = temp/10;
58     }
59     if (temp>0)
```



```

52     {
53         s.insert(0,1,numToString(temp));
54     }
55     return s;
56 }
57 string nhan(string a, string b)    // nhan 2 so lon
58 {
59     string s = "";
60     int l = a.length();
61     string s1[l];
62     for (int i=l-1; i>=0; i--)    // nhan tung chu so cua a voi b sau do cong don vao
63     {
64         s1[i] = nhanNho(a[i], b);    // nhan tung so cua a voi b
65         s1[i].insert(s1[i].length(), 1-i-1, '0');
66         s = sum(s, s1[i]);    // cong don theo cach cong so lon
67     }
68     return s;
69 }
70 int main(int argc, char **argv)
71 {
72     string a, b, s;
73     cout<<"Nhap a va b"<<endl;
74
75     getline(cin, a);
76     getline(cin, b);
77
78     s = sum(a,b);
79     cout<<"Tong cua a va b : "<<s<<endl;
80
81     s = nhan(a,b);
82     cout<<"Tich cua a va b : "<<s<<endl;
83     return 0;
84 }

```

```

Nhap a va b
123456789101112131415
2019181716151413121110
Tong cua a va b : 2142638505252525252525
Tich cua a va b : 249281691287726668778975627329629630670650

-----
(program exited with code: 0)
Press return to continue

```

Bài 19: Đổi 1 số hệ 10 sang hệ 2 và ngược lại.

Bài 20: Đổi 1 số hệ 10 sang hệ 16 và ngược lại.

Bài 21: Đổi 1 số hệ 16 sang hệ 2 và ngược lại.

```

/*
  Bài 19-20-21
*/
#include<bits/stdc++.h>
#include<string>
#define ll long long
using namespace std;
string st;
string Dec2Bin(ll x){
  string s="";
  while(x>0){
    s=char(x%2+48)+s; //char(x%2+48) >> Chuyển số dư phép chia x cho 2 ra kí tự số nên +48
    x>>=1; //Dịch 1 bit sang phải <=> Chia x cho 2
  }
  return s;}
ll Bin2Dec(string s){
  ll kq=0;
  for(int i=0;i<s.length();i++){
    //kq+=int(s[i]-48)*pow(2,s.length()-i-1); //int(s[i]-48) đổi ra số
    kq+=int(s[i]-48)*(1<<(s.length()-i-1)); //1<<k --> Dịch sang trái k bit (2^k)
  }
  return kq;}
ll Hex2Dec(string s){
  ll so,kq=0;
  for(int i=0;i<s.length();i++){
    so=(int(s[i])>=65 && int(s[i])<=70)? int(s[i]-55):int(s[i]-48);
    //kq+=so*16<<(4*(s.length()-i-1)-4); thiếu TH cuối
    kq+=so*pow(16,s.length()-i-1);
  }
  return kq;}
int main(){
  ll nd; string nb,nh,nx;
  freopen("BIT.INP","r",stdin);
  freopen("BIT.OUT","w",stdout);
  ios_base::sync_with_stdio(false);
  cin.tie(0);cout.tie(0);
  cin>>nd;
  cin>>nb;
  cin>>nh;
  cout<<Dec2Bin(nd)<<endl;
  cout<<Bin2Dec(nb)<<endl;
  cout<<Hex2Dec(nh)<<endl;
  /*
  int d=0;
  while(getline(cin,nx)){
    d++;
    if(d==1){
      nd=stoi(nx);
      cout<<nd<<endl;
    } else cout<<nx<<endl;
  } //cout<<Dec2Bin(nd);
  */
  return 0;}
/*
cout<<(16<<-4)<<endl; //4*0-4 >>16^0
cout<<(16<<0)<<endl; //4*1-4 >>16^1
cout<<(16<<4)<<endl; //4*2-4 >>16^2
cout<<(16<<8)<<endl; //4*3-4 >>16^3
cout<<(16<<12)<<endl; //4*4-4 >>16^4
*/

```

Còn nữa...