

NGÀY – DATE

- Nguyễn Hoàng Yến Như
 - Nguyễn Trần Phúc Nghi
 - Nguyễn Trần Phúc An
 - Nguyễn Đức Anh Phúc
- Trịnh Thị Thanh Trúc
 - KS. Hồ Thái Ngọc
 - KS. Cao Bá Kiệt
 - KS. Lê Ngọc Huy
 - CN. Bùi Cao Doanh
 - CN. Nguyễn Trọng Thuận
 - KS. Phan Vĩnh Long
 - KS. Nguyễn Cường Phát
 - ThS. Nguyễn Hoàng Ngân
- ThS. Đỗ Văn Tiến
 - ThS. Nguyễn Hoàn Mỹ
 - ThS. Dương Phi Long
 - ThS. Trương Quốc Dũng
 - ThS. Nguyễn Thành Hiệp
 - ThS. Nguyễn Võ Đăng Khoa
 - ThS. Võ Duy Nguyên
 - ThS. Trần Việt Thu Phương
 - TS. Nguyễn Tấn Trần Minh Khang

Khai báo kiểu dữ liệu

- Kiến thức phổ thông: ngày có ba thành phần thông tin: ngày, tháng, năm.
- Năm là số nguyên dương lớn hơn bằng 1.
- Tháng là số nguyên dương nằm trong đoạn $[1,12]$.
- Ngày là số nguyên dương nằm trong đoạn $[1,x]$ với $x \in \{28,29,30,31\}$.
- Ngày đầu tiên theo dương lịch là ngày 01/01/01 và ngày này là ngày thứ hai.

Khai báo kiểu dữ liệu

— Khai báo kiểu dữ liệu

```
101.struct Ngay
102.{
103.|    int Ngay;
104.|    int Thang;
105.|    int Nam;
106.};
107.typedef struct Ngay NGAY;
```

Nhập ngày

— Định nghĩa hàm

```
101. void Nhap(NGAY &x)
102. {
103.     cout << "Nhap ngay: ";
104.     cin >> x.Ngay;
105.     cout << "Nhap thang: ";
106.     cin >> x.Thang;
107.     cout << "Nhap nam: ";
108.     cin >> x.Nam;
109. }
```

Xuất ngày

```
101. void Xuat(NGAY x)
102. {
103.     cout << "\nNgày:" << x.Ngay;
104.     cout << "\nThang:" << x.Thang;
105.     cout << "\nNam:" << x.Nam;
106. }
107. void Xuat(NGAY x)
108. {
109.     cout << x.Ngay << "/" << x.Thang << "/" << x.Nam;
110. }
```

Kiểm tra năm nhuận

- Một năm được gọi là năm nhuận khi thoả một trong hai điều kiện.
 - + Điều kiện 1: Năm chia hết cho 4 và không chia hết cho 100.
 - + Điều kiện 2: Năm chia hết cho 400.
- Ví dụ 1:
 - + Dữ liệu vào: 1996.
 - + Dữ liệu ra: Nhuận.
 - + Lưu ý: Nhuận theo điều kiện 1.

Kiểm tra năm nhuận

- Một năm được gọi là năm nhuận khi thoả một trong hai điều kiện.
 - + Điều kiện 1: Năm chia hết cho 4 và không chia hết cho 100.
 - + Điều kiện 2: Năm chia hết cho 400.
- Ví dụ 2:
 - + Dữ liệu vào: 2000.
 - + Dữ liệu ra: Nhuận.
 - + Lưu ý: Nhuận theo điều kiện 2.

Kiểm tra năm nhuận

- Một năm được gọi là năm nhuận khi thoả một trong hai điều kiện.
 - + Điều kiện 1: Năm chia hết cho 4 và không chia hết cho 100.
 - + Điều kiện 2: Năm chia hết cho 400.
- Ví dụ 3:
 - + Dữ liệu vào: 1900.
 - + Dữ liệu ra: Không nhuận.

Kiểm tra năm nhuận

- Một năm được gọi là năm nhuận khi thoả một trong hai điều kiện.
 - + Điều kiện 1: Năm chia hết cho 4 và không chia hết cho 100.
 - + Điều kiện 2: Năm chia hết cho 400.
- Hàm `ktNhuan` trả về một trong hai giá trị.
 - + Giá trị 1: năm nhuận.
 - + Giá trị 0: năm không nhuận.

Kiểm tra năm nhuận

— Định nghĩa hàm

```
101.int ktnhuan(NGAY x)
102.{
103.    if(x.Nam % 4 == 0 && x.Nam % 100 !=0)
104.        return 1;
105.    if(x.Nam % 400 == 0)
106.        return 1;
107.    return 0;
108.}
```

Tìm số ngày tối đa trong tháng

— Kiến thức phổ thông

- + Các tháng có 31 ngày: 1, 3, 5, 7, 8, 10, 12.
- + Các tháng có 30 ngày: 4, 6, 9, 11.
- + Tháng 2
 - có 28 ngày (năm không nhuận)
 - 29 ngày (năm nhuận).

Tìm số ngày tối đa trong tháng

— Định nghĩa hàm

```
101.int SoNgayToiDaTrongThang(NGAY x)
102.{
103.    int ngaythang[12] = {31,28,31,30,31,30,31,
104.                           31,30,31,30,31};
105.    if(ktNhuan(x)==1)
106.        ngaythang[1] = 29;
107.    return ngaythang[x.Thang-1];
108.}
```

Tìm số ngày tối đa trong năm

— Kiến thức phổ thông

+ Năm nhuận có 366 ngày.

+ Năm không nhuận có 365 ngày.

— Định nghĩa hàm

```
101.int SoNgayToiDaTrongNam(NGAY x)
102.{
103.    if(ktNhuan(x)==1)
104.        return 366;
105.    return 365;
106.}
```

Kiểm tra tính hợp lệ của một ngày

- Một ngày được gọi là hợp lệ khi thỏa các điều kiện:
 - + Năm hợp lệ: năm lớn hơn 1.
 - + Tháng hợp lệ: tháng nằm trong đoạn [1,12].
 - + Ngày hợp lệ: ngày lớn hơn 1 và nhỏ hơn bằng số ngày tối đa của tháng và năm hợp lệ.
- Hàm `ktHopLe` trả về một trong hai giá trị.
 - + Giá trị 1: ngày hợp lệ.
 - + Giá trị 0: ngày không hợp lệ.

Kiểm tra tính hợp lệ của một ngày

```
101.int ktHopLe(NGAY x)
102.{
103.    if(x.Nam < 1)
104.        return 0;
105.    if(x.Thang < 1)
106.        return 0;
107.    if(x.Thang > 12)
108.        return 0;
109.    if(x.Ngay < 1)
110.        return 0;
111.    if(x.Ngay > SoNgayToiDaTrongThang(x))
112.        return 0;
113.    return 1;
114.}
```

- Một ngày được gọi là hợp lệ khi thỏa các điều kiện:
 - + Năm hợp lệ: năm lớn hơn 1.
 - + Tháng hợp lệ: tháng nằm trong đoạn [1,12].
 - + Ngày hợp lệ: ngày lớn hơn 1 và nhỏ hơn bằng số ngày tối đa của tháng và năm hợp lệ.
- Hàm trả về một trong hai giá trị.
 - + Giá trị 1: ngày hợp lệ.
 - + Giá trị 0: ngày không hợp lệ.

So sánh hai ngày

— Qui ước:

Quá khứ < Hiện tại < Tương lai

— Giá trị trả về: Hàm **SoSanh** trả về một trong 3 giá trị: +1, 0, -1.

+ Giá trị -1: ngày thứ nhất nhỏ hơn ngày thứ hai.

+ Giá trị 0: ngày thứ nhất bằng ngày thứ hai.

+ Giá trị +1: ngày thứ nhất lớn hơn ngày thứ hai.

So sánh hai ngày

```
101.int SoSanh(NGAY x,NGAY y)
102.{
103.    if(x.Nam > y.Nam)
104.        return 1;
105.    if(x.Nam < y.Nam)
106.        return -1;
107.    if(x.Thang > y.Thang)
108.        return 1;
109.    if(x.Thang < y.Thang)
110.        return -1;
111.    if(x.Ngay > y.Ngay)
112.        return 1;
113.    if(x.Ngay < y.Ngay)
114.        return -1;
115.    return 0;
116.}
```

Tính số thứ tự của ngày trong năm

— Ví dụ: 20/10/2007.

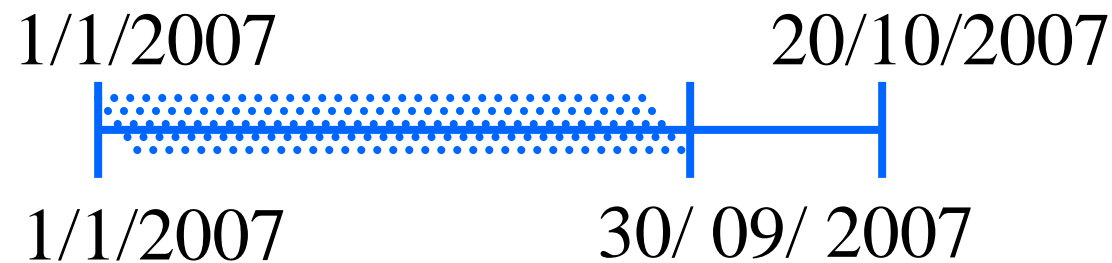
— Kết quả: $31_1 + 28_2 + 31_3 + 30_4 + 31_5 + 30_6 + 31_7 + 31_8 + 30_9 + 20_{10} = 293$.

Tháng	01	02	03	04	05	06	07	08	09
Số ngày tối đa	31	28	31	30	31	30	31	31	30

— Như vậy: Ngày 20/10/2007 có số thứ tự ngày trong năm là 293.

Tính số thứ tự của ngày trong năm

- Việc tính số thứ tự của ngày trong năm thực hiện qua hai giai đoạn.
 - + Giai đoạn 01: tính tổng số ngày trong những tháng đã qua.
 - + Giai đoạn 02: tính ngày trong tháng hiện hành.

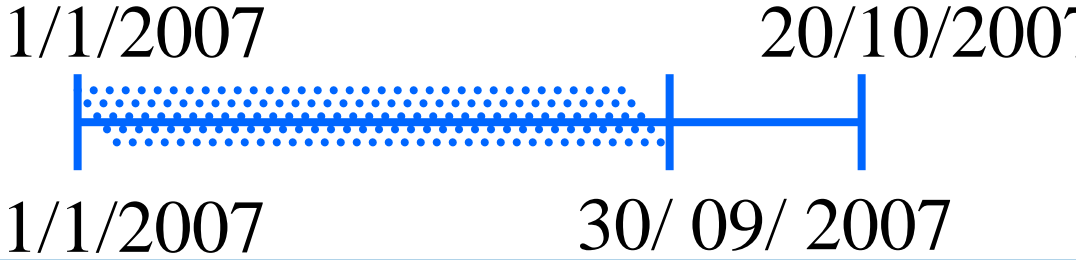


Tính số thứ tự của ngày trong năm

```

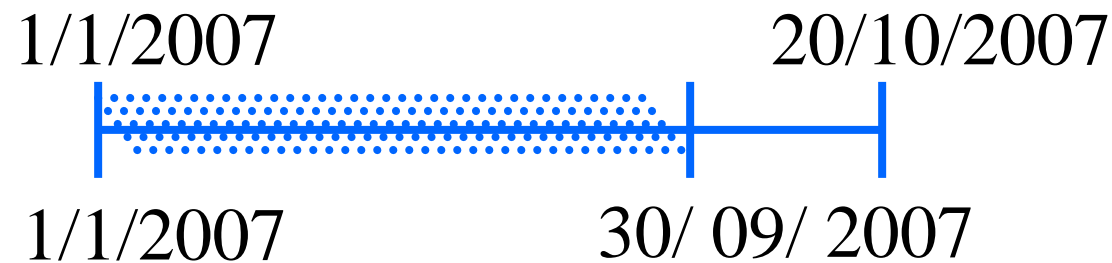
101.int SoThuTuTrongNam(NGAY x)
102.{
103.    int stt = 0;
104.    for(int i=1; i<=x.Thang-1; i++)
105.    {
106.        |    NGAY temp = {1,i,x.Nam};
107.        |    stt = stt + SoNgayToiDaTrongThang(temp);
108.    }
109.    return (stt + x.Ngay);
110.}

```



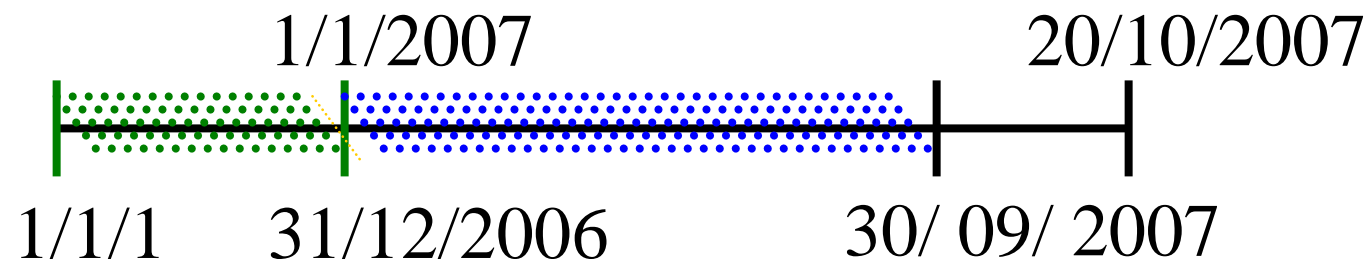
Tính số thứ tự của ngày trong năm

- Việc tính số thứ tự của ngày trong năm thực hiện qua hai giai đoạn.
 - + Giai đoạn 01: tính tổng số ngày trong những tháng đã qua.
 - + Giai đoạn 02: tính ngày trong tháng hiện hành.



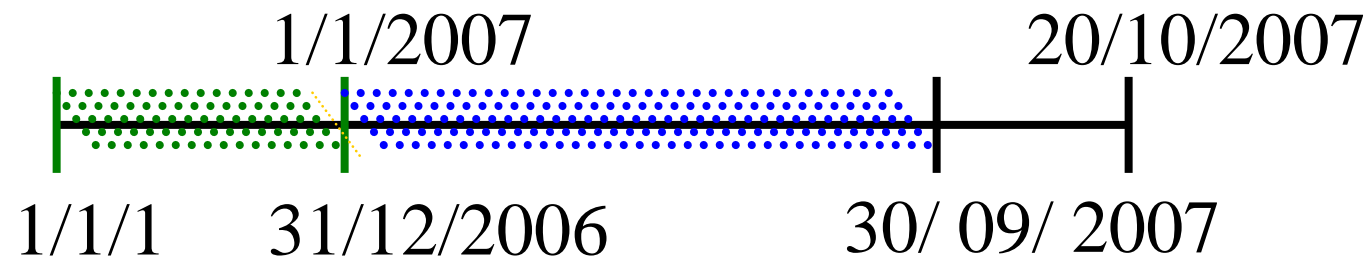
Tính số thứ tự ngày kể từ ngày 1/1/1

- Ví dụ: 20/10/2007.
- Kết quả: $365_1 + 365_2 + 365_3 + 366_4 + \dots + 293_{2007} = 734,259$.
- Như vậy: Ngày 20/10/2007 có số thứ tự ngày kể từ ngày 1/1/1 là 734,259.



Tính số thứ tự ngày kể từ ngày 1/1/1

- Việc tính số thứ tự của ngày kể từ ngày 01/01/01 thực hiện qua hai giai đoạn.
 - + Giai đoạn 01: tính tổng số ngày trong những năm đã qua (trong ví dụ trên ta tính tổng số ngày từ năm 01 tới năm 2006).
 - + Giai đoạn 02: tính số thứ tự ngày trong năm hiện hành (năm 2007).

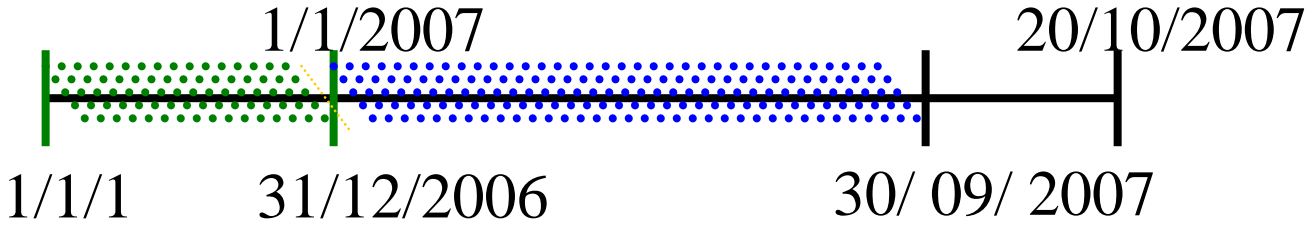


Tính số thứ tự ngày kể từ ngày 1/1/1.

```

101.int SoThuTu(NGAY x)
102.{
103.    int stt = 0;
104.    for(int i=1; i<=x.Nam-1; i++)
105.    {
106.        NGAY temp = {1,1,i};
107.        stt = stt + SoNgayToiDaTrongNam(temp);
108.    }
109.    return (stt + SoThuTuTrongNam(x));
110.}

```



Xuất thứ của ngày

- Ngày đầu tiên theo dương lịch là ngày 01/01/01.
- Ngày 01/01/01 là ngày thứ hai có stt ngày là 1 và $1 \bmod 7 = 1$.
- Suy ra 02/01/01 là ngày thứ ba có stt ngày là 2 và $2 \bmod 7 = 2$.
- Suy ra 03/01/01 là ngày thứ tư có stt ngày là 3 và $3 \bmod 7 = 3$.
- Suy ra 04/01/01 là ngày thứ năm có stt ngày là 4 và $4 \bmod 7 = 4$.
- Suy ra 05/01/01 là ngày thứ sáu có stt ngày là 5 và $5 \bmod 7 = 5$.
- Suy ra 06/01/01 là ngày thứ bảy có stt ngày là 6 và $6 \bmod 7 = 6$.
- Suy ra 07/01/01 là ngày chủ nhật có stt ngày là 7 và $7 \bmod 7 = 0$.
- Suy ra 08/01/01 là ngày thứ hai có stt ngày là 8 và $8 \bmod 7 = 1$.
- Các thứ trong tuần có chu kỳ là 7 ngày.

```
101. void XuatThu(NGAY x)
102. {
103.     int stt = SoThuTu(x);
104.     switch(stt%7)
105.     {
106.         case 0: cout << "Chủ Nhật";
107.             break;
108.         case 1: cout << "Thứ Hai";
109.             break;
110.         case 2: cout << "Thứ Ba";
111.             break;
112.         case 3: cout << "Thứ Tư";
113.             break;
114.         case 4: cout << "Thứ Năm";
115.             break;
116.         case 5: cout << "Thứ Sáu";
117.             break;
118.         case 6: cout << "Thứ Bảy";
119.             break;
120.     }
121. }
```

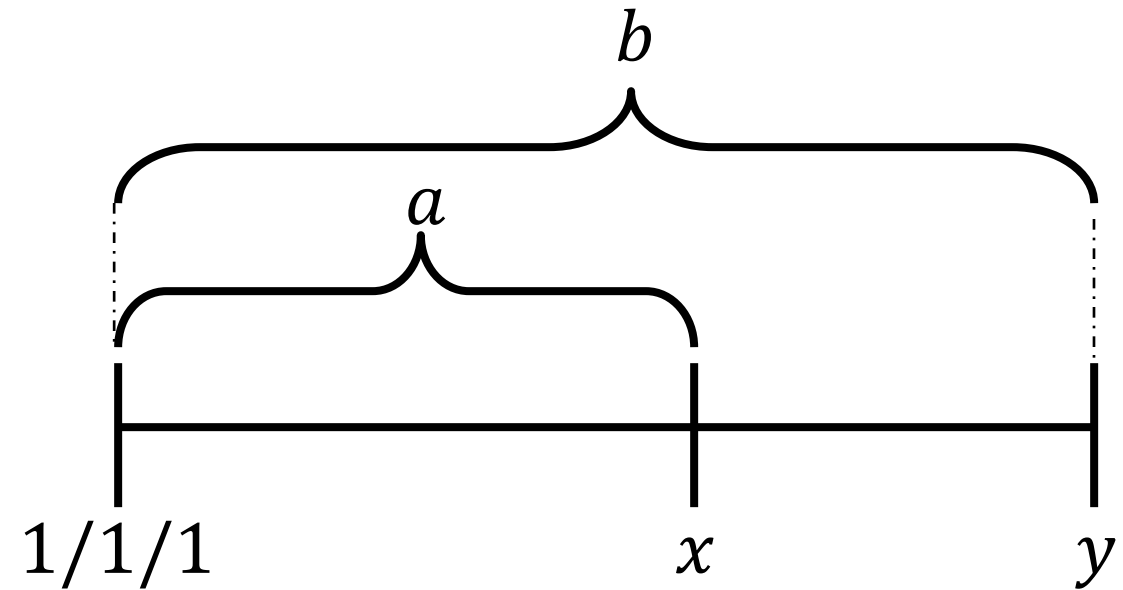
Khoảng cách giữa hai ngày

— Định nghĩa hàm

```

101.int  KhoangCach(NGAY x,NGAY y)
102.{
103.    int a = SoThuTu(x);
104.    int b = SoThuTu(y);
105.    return abs(a-b);
106.}

```



Tìm ngày khi biết năm và stt

- Ví dụ: năm=2002, stt=178.
- Kết quả: ???
- Cách làm:
- Kết quả: 27/06/2002.

Tháng = 1	178-31 > 0 Đ
stt = 147	
Tháng = 2	147-28 > 0 Đ
stt = 119	
Tháng = 3	119-31 > 0 Đ
stt = 88	
Tháng = 4	88-30 > 0 Đ
stt = 58	
Tháng = 5	58-31 > 0 Đ
stt = 27	
Tháng = 6	27-30 > 0 S
Ngày = 27	
Tháng = 6	

Tìm ngày khi biết năm và stt

```

101. NGAY TimNgay(int Nam, int stt)
102. {
103.     NGAY temp = {1, 1, Nam};
104.     temp.Thang = 1;
105.     while(stt - SoNgayToiDaTrongThang(temp) > 0)
106.     {
107.         stt = stt - SoNgayToiDaTrongThang(temp);
108.         temp.Thang++;
109.     }
110.     temp.Ngay = stt;
111.     return temp;
112. }

```

Tháng = 1	178-31 > 0 Đ
stt = 147	
Tháng = 2	147-28 > 0 Đ
stt = 119	
Tháng = 3	119-31 > 0 Đ
stt = 88	
Tháng = 4	88-30 > 0 Đ
stt = 58	
Tháng = 5	58-31 > 0 Đ
stt = 27	
Tháng = 6	27-30 > 0 S
Ngày = 27	
Tháng = 6	

Tìm ngày khi biết stt

- Ví dụ: stt = 1734
- Kết quả: ???
- Cách làm:
- Kết quả trung gian: năm 5, stt = 273.

Năm = 1	
stt = 1369	1734-365 > 0 Đ
Năm = 2	
stt = 1004	1369-365 > 0 Đ
Năm = 3	
stt = 639	1004-365 > 0 Đ
Năm = 4	
stt = 273	639-366 > 0 Đ
Năm = 5	
stt = 273	273-365 > 0 S
Năm = 5	

Tìm ngày khi biết stt

```

101. NGAY TimNgay(int stt)
102. {
103.     int Nam = 1;
104.     int sn = 365;
105.     while(stt - sn > 0)
106.     {
107.         stt = stt - sn;
108.         Nam++;
109.         NGAY temp = {1, 1, Nam};
110.         sn = SoNgayToiDaTrongNam(temp);
111.     }
112.     return TimNgay(Nam, stt);
113. }

```

Năm = 1	1734 - 365 > 0 Đ
stt = 1369	
Năm = 2	1369 - 365 > 0 Đ
stt = 1004	
Năm = 3	1004 - 365 > 0 Đ
stt = 639	
Năm = 4	639 - 366 > 0 Đ
stt = 273	
Năm = 5	273 - 365 > 0 S
stt = 273	
Năm = 5	

Ngày kế tiếp

- Ví dụ: ngày cần tìm ngày kế tiếp là 12/07/2019.
- Kết quả: 13/07/2019.
- Thuật toán
 - + Gọi ngày cần tìm ngày kế tiếp là ngày x .
 - + Tính số thứ tự ngày kể từ ngày 01/01/01 tới ngày x (gọi là stt).
 - + Ngày kế tiếp của ngày x sẽ có số thứ tự ngày kể từ ngày 01/01/01 là $stt + 1$.
 - + Tìm ngày khi biết số thứ tự ngày kể từ ngày 01/01/01.

Ngày kế tiếp

— Định nghĩa hàm

```
101. NGAY KeTiep(NGAY x)
102. {
103.     int stt = SoThuTu(x);
104.     stt = stt + 1;
105.     return TimNgay(stt);
106. }
```

Tìm ngày hôm qua

— Định nghĩa hàm

```
101. NGÀY TruocDo(NGÀY x)
102. {
103.     if(x.Ngay == 1 && x.Thang == 1 && x.Nam == 1)
104.         return x;
105.     int stt = SoThuTu(x);
106.     stt = stt - 1;
107.     return TimNgay(stt);
108. }
```

Tìm ngày kế đó k ngày

— Định nghĩa hàm

```
101. NGÀY KeTiep(NGÀY x, int k)
102. {
103.     int stt = SoThuTu(x);
104.     stt = stt + k;
105.     return TimNgay(stt);
106. }
```

Tìm ngày trước đó k ngày

— Định nghĩa hàm

```
101. NGAY TruocDo(NGAY x, int k)
102. {
103.     NGAY temp = x;
104.     for(int i=1; i<=k; i++)
105.         temp = TruocDo(temp);
106.     return temp;
107. }
```

Chúc các bạn học tốt
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN TP.HCM

Nhóm UIT-Together
Nguyễn Tấn Trần Minh Khang