

## ĐỀ KIỂM TRA GIỮA KỲ # HK1 – 2021-2022

Môn: **Nhập môn Lập trình** - Lớp: **21CTT3** - GV: **Thái Hùng Văn**

Thời gian làm bài: **75 phút** - ***Không*** dùng tài liệu, ĐTDĐ, bàn phím Laptop /Desktop PC

### Câu 1:

Hãy viết hàm F để dòng code bên dưới có kết quả  $X=11$  và  $Y=2021$  :

a/ `int X, Y = X + F(X);`

b/ `int X, Y = X + Y - F(X, Y);`

### Câu 2:

Vẽ lưu đồ (Flow Chart) kiểm tra một số có phải là số nguyên tố không.

### Câu 3:

Gọi ***thời\_điểm*** là một mốc giờ:phút:giây cụ thể trong ngày và ***thời\_gian*** là số giây giữa 2 ***thời\_điểm***.

(vd, ***thời\_điểm*** bắt đầu làm bài thi là 09:30:00 và ***thời\_điểm*** kết thúc là 10:30:01 thì ***thời\_gian*** thi là 3601 giây)

Hãy viết các hàm thực hiện các việc sau :

a/ Xác định ***thời\_gian*** đã trôi qua kể từ **00:00:00** đến một ***thời\_điểm*** nào đó.

(vd, ***thời\_gian*** đã trôi qua từ 0 giờ đến 01:02:03 là 3723 giây)

b/ Xác định ***thời\_gian*** đã trôi qua giữa 02 ***thời\_điểm***

c/ Xác định ***thời\_điểm*** khi biết ***thời\_gian*** đã trôi qua kể từ **00:00:00**.

d/ Sắp xếp lại 3 ***thời\_điểm*** đưa vào theo thứ tự tăng dần.

-- HẾT --

# Bài Sửa

## CÂU 1

a/ viết hàm F để dòng code {**int X, Y = X + F(X);**} có kết quả X=11 và Y=2021 :

\* Nếu biểu thức **Y=X+F(X)** thực hiện từ trái qua phải (nhớ giá trị X rồi mới tính F và cập nhật X) :

```
int F ( int & a )
{
    int temp = a ;
    a = 11 ;
    return 2021-temp ;
}
```

\* Nếu biểu thức **Y=X+F(X)** thực hiện từ phải qua trái (tính F và cập nhật X rồi mới tính tổng) :

```
int F ( int & a )
{
    a = 11 ;
    return 2021-a ;
}
```

b/ viết hàm F để dòng code {**int X, Y = X + Y - F(X, Y);**} có kết quả X=11 và Y=2021 :

\* Nếu **Y=X+Y+F(X,Y)** thực hiện từ trái qua phải (nhớ giá trị X rồi mới tính F và cập nhật X) :

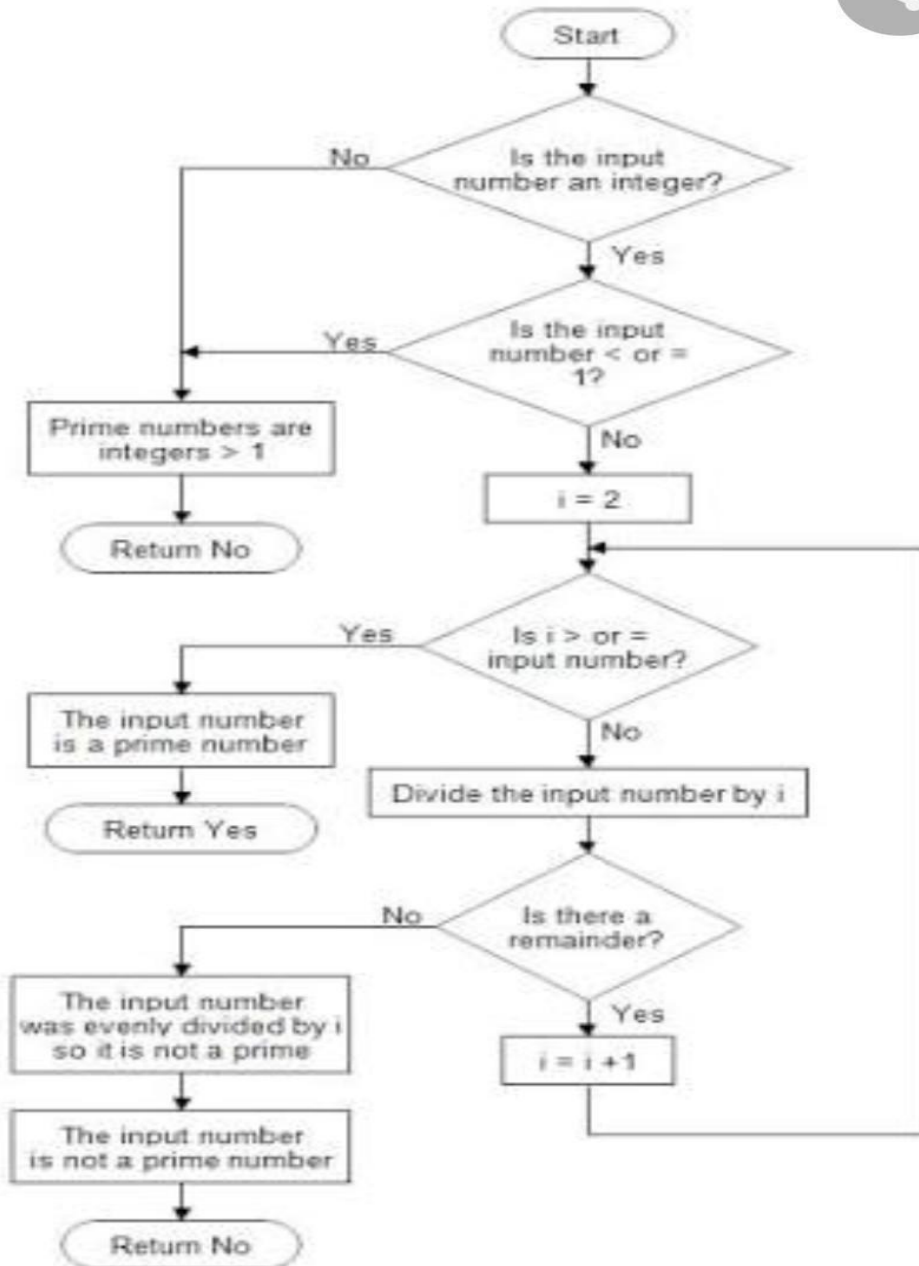
```
int F ( int & a, int b ) { int temp = a ; a = 11 ; return temp+b-2021 ; }
```

\* Nếu **Y=X+Y+F(X,Y)** thực hiện từ phải qua trái (tính F và cập nhật X rồi mới tính tổng) :

```
int F ( int & a, int b ) {a = 11 ; return a+b-2021 ; }
```

## CÂU 2 :

## Function: IsThisNumberPrime

**CÂU 3**

\* **Prototype 4 hàm của 4 câu như sau :**

**int ThoiGian ( int hh, int mm, int ss ) ;** // hàm trả về số giây đã trôi qua từ thời\_điểm 00:00:00 đến hh:mm:ss

**int ThoiGian ( int h1, int m1, int s1, int h2, int m2, int s2 ) ;** // hàm trả về số giây từ thời\_điểm h1:m1:s1 đến h2:m2:s2

**void ThoiDiem ( int seconds, int &hh, int &mm, int &ss ) ;** // Xác định thời\_điểm hh:mm:ss khi biết số giây kể từ 0 giờ

**int GanNhat ( int h1, int m1, int s1, int h2, int m2, int s2, int h3, int m3, int s3 ) ;** // Xác định 2 thời\_điểm gần nhau nhất trong 3 thời\_điểm T1, T2, T3. Giá trị trả về của hàm là 12 khi kết quả là T1 & T2, là 13 khi kết quả là T1 & T3, là 23 khi kết quả là T2 & T3

a/ Xác định *thời\_gian* đã trôi qua kể từ **00:00:00** đến một *thời\_điểm* nào đó.

(vd, *thời\_gian* đã trôi qua từ 0 giờ đến 01:02:03 là 3723 giây)

b/ Xác định *thời\_gian* đã trôi qua giữa 02 *thời\_điểm*

c/ Xác định *thời\_điểm* khi biết *thời\_gian* đã trôi qua kể từ **00:00:00**.

d/ Sắp xếp lại 3 *thời\_điểm* đưa vào theo thứ tự tăng dần.

### \* Thân hàm :

```
int ThoiGian ( int hh, int mm, int ss ) {
```

```
    return hh*3600 + mm*60 + ss ;
```

```
}
```

```
int ThoiGian ( int h1, int m1, int s1 , int h2, int m2, int s2) {
```

```
    int ss1 = ThoiGian (h1, m1, s1) ; // số giây từ 0 giờ đến th/đ h1:m1:s1 (cũng có thể coi là th/đ theo giây)
```

```
    int ss2 = ThoiGian (h2, m2, s2) ; // số giây từ 0 giờ đến th/đ h2:m2:s2 (cũng có thể coi là th/đ theo giây)
```

```
    if (ss2 > ss1 )
```

```
        return ss2 – ss1 ;
```

```
    return ss1 - ss2 ;
```

```
}
```

```
void ThoiDiem ( int seconds, int &hh, int &mm, int &ss) {
```

```
    hh = seconds / 3600 ; // do 1 giờ = 3600 giây
```

```
    int s = seconds % 3600 ; // số giây đã trôi qua tính từ đầu giờ
```

```
    mm = s / 60 ;
```

```
    ss = s % 60;
```

```
}
```

```
int GanNhat ( int h1, int m1, int s1, int h2, int m2, int s2, int h3, int m3, int s3 ) {
```

```
    int t12 = ThoiGian (h1, m1, s1, h2, m2, s2) ; // số giây từ giữa 2 th/đ T1 & T2
```

```
    int t13 = ThoiGian (h1, m1, s1, h3, m3, s3) ; // số giây từ giữa 2 th/đ T1 & T3
```

```
    int t23 = ThoiGian (h2, m2, s2, , h3, m3, s3) ; // số giây từ giữa 2 th/đ T2 & T3
```

```
    if (t12< t13 && t12<t23) return 12 ;
```

```
    if (t13< t12 && t13<t23) return 13 ;
```

```
    return 23;
```

```
}
```

### \* CT minh họa việc dùng các hàm trên:

```

int main ( ) {
    cout << "\n So giay tu 0 gio den 7:30 la : " << ThoiGian (7, 30, 0) ;
    cout << "\n So giay tu 7:30 den 9:10 la : " << ThoiGian (7,30,0, 9,10,0) ;
    int h, m, s, t = 2021 ;
    ThoiDiem (t, h, m, s) ;
    cout<< "\n Thoi diem "<< t << " giay (ke tu 0 giờ) la : " << h << ":" << m << ":" << s;
    cout << "\n Voi 3 th/d T1=12:34:56, T2=01:02:03, T3=21:00:00 thi 2 th/d gan nhat la : "
        << GanNhat (12,34,56, 1,2,3, 21,0,0) ;
}

```

@ Nếu đã học kiểu Struct thì nên làm như sau :

\* Định nghĩa kiểu ThoiDiem :

```

struct THOIDIEM {
    int hh, mm, ss ;
};

```

\* Prototype 4 hàm :

**int ThoiGian ( THOIDIEM T ) ;** // hàm trả về số giây đã trôi qua từ thời\_điểm 00:00:00 đến thời\_điểm T

**int ThoiGian ( THOIDIEM T1, THOIDIEM T2 ) ;** // hàm trả về số giây từ thời\_điểm T1 đến T2

**THOIDIEM ThoiDiem ( int seconds ) ;** // Xác định thời\_điểm khi biết số giây kể từ 0 giờ

**int GanNhat ( THOIDIEM T1, THOIDIEM T2, THOIDIEM T3 ) ;** // Xác định 2 thời\_điểm gần nhau nhất trong 3 thời\_điểm T1, T2, T3. Giá trị trả về của hàm là 12 khi kết quả là T1 & T2, là 13 khi kq là T1 & T3, là 23 khi kq là T2 & T3

\* Thân hàm :

```

int ThoiGian ( THOIDIEM T) {
    return T.hh*3600 + T.mm*60 + T.ss ;
}

```

```

int ThoiGian ( THOIDIEM T1, THOIDIEM T2) {
    int ss1 = ThoiGian (T1) ; // số giây từ 0 giờ đến th/đ T1
    int ss2 = ThoiGian (T2) ; // số giây từ 0 giờ đến th/đ T2
    if (ss2 > ss1 ) return ss2 – ss1 ;
    return ss1 - ss2 ;
}

```

```
}
```

```
THOIDIEM ThoiDiem ( int seconds) {  
    THOIDIEM T ;  
    T.hh = seconds / 3600 ; // do 1 giờ = 3600 giây  
    int s = seconds % 3600 ; // số giây đã trôi qua tính từ đầu giờ  
    T.mm = s / 60 ;  
    T.ss = s % 60;  
    return T ;  
}
```

```
}
```

```
int GanNhat ( THOIDIEM T1, THOIDIEM T2, THOIDIEM T3 ) {  
    int t12 = ThoiGian (T1, T2) ; // số giây từ giữa 2 th/d T1 & T2  
    int t13 = ThoiGian (T1, T3) ; // số giây từ giữa 2 th/d T1 & T3  
    int t23 = ThoiGian (T2, T3) ; // số giây từ giữa 2 th/d T2 & T3  
    if (t12< t13 && t12<t23) return 12 ;  
    if (t13< t12 && t13<t23) return 13 ;  
    return 23;  
}
```

\* CT minh họa việc dùng các hàm trên:

```
int main ( ) {  
    THOIDIEM T1 = {12,34,56} , T2 = {1,2,3} , T3 = {21,0,0} ;  
    cout << « Xet cac th/d T1, T2, T3 la : »; XuatThoiDiem(T1); XuatThoiDiem(T2); XuatThoiDiem(T3);  
    cout << "\n So giay tu 0 gio den T1 la : " << ThoiGian (T1) ;  
    cout << "\n So giay giua 2 th/d T1 & T2 la : " << ThoiGian (T1,T2) ;  
    int t = 2021 ;  
    THOIDIEM T = ThoiDiem (t) ;  
    cout<< "\n Thoi diem ung voi " << t << " giay (ke tu 0 giờ) la : " ; XuatThoiDiem(T);  
    cout << "\n Voi 3 th/d T1, T2, T3 tren thi 2 th/d gan nhat la : " << GanNhat (T1, T2, T3) ;  
}
```