



ĐẠI HỌC KINH TẾ KỸ THUẬT CÔNG NGHIỆP
KHOA CÔNG NGHỆ THÔNG TIN

Bài Thực Tập

Giải bài tập bằng sử dụng hàm

Hà Nội – 2018

NỘI DUNG

1

Mục tiêu bài học

2

Hướng dẫn học tập

3

Nội dung bài học

4

Giao nhiệm vụ tuần tiếp theo

MỤC TIÊU BÀI HỌC

- **Mục tiêu:**

Trang bị cho sinh viên kỹ năng lập trình cơ bản trong C++:

- + Khai báo và định nghĩa hàm
- + Lời gọi hàm
- + Viết chương trình sử dụng hàm
- + Phát hiện và sửa lỗi

- **Kết quả đạt được:**

Sinh viên thành thạo cách sử dụng hàm áp để giải các bài tập từ đơn giản đến phức tạp.

HƯỚNG DẪN HỌC TẬP

Để hoàn thành tốt bài học này sinh viên cần thực hiện những nhiệm vụ sau:

- Đọc trước tài liệu: “B5_Tailieu_TTLTCB” phần nhắc lại kiến thức lý thuyết mục “E. TÓM TẮT LÝ THUYẾT”
- Cài đặt, sử dụng được công cụ thực hành Cfree 5.0.
- Thực hành trên máy tính các bài thực hành mẫu.
- Hoàn thành các bài thực hành tự làm cuối bài học.
- Hoàn thành các bài tập giao về nhà.
- Trao đổi, thảo luận với giảng viên qua các phương thức:
 - + Thảo luận đặt câu hỏi trên diễn đàn.

NỘI DUNG BÀI HỌC

I. Hướng dẫn ban đầu (90 phút)

- 1.1 Hướng dẫn cài đặt công cụ học tập
- 1.2 Tiến trình bài thực hành
- 1.3 Hướng dẫn thực hành bài số 1
- 1.4 Hướng dẫn thực hành bài số 2

II. Hướng dẫn thường xuyên (90 phút)

- 2.1 Hướng dẫn thực hành bài số 03
- 2.2 Hướng dẫn thực hành bài số 04

III. Bài tập tự giải (90 phút)

I. HƯỚNG DẪN BAN ĐẦU

- Sinh viên cần cài đặt được công cụ thực hành.
- Yêu cầu sinh viên đọc lại và ghi nhớ các kiến thức lý thuyết trong mục E trong tài liệu “*B5_Tailieu_TTLTCB*”
- Hướng dẫn chi tiết ví dụ mẫu để hiểu rõ về cách sử dụng hàm.

I. HƯỚNG DẪN BAN ĐẦU

Cách 1

- Định hướng tiền xử lý.
- Khai báo nguyên mẫu hàm
- Hàm main()
- Định nghĩa hàm

Cách 2

- Định hướng tiền xử lý.
- Định nghĩa hàm
- Hàm main()

I. HƯỚNG DẪN BAN ĐẦU

Khai báo nguyên mẫu hàm

- Cú pháp:

Kiểu_trả_về Tên_hàm(Danh sách kiểu tham số);

Ví dụ:

long square(int);

float sum(float, float, float);

int maximize(int, int, int);

I. HƯỚNG DẪN BAN ĐẦU

Định nghĩa hàm

- Cú pháp:

```
Kiểu_trả_về    Tên_hàm(Danh sách tham số)
{
    khai_báo_biến_địa_phương;
    lệnh_trong_thân_hàm;
    return giá_trị_hồi_trả;
}
```

```
long square(int x)
{
    return x*x;
}
```

I. HƯỚNG DẪN BAN ĐẦU

```
float  sum(float  x, float y, float z)
{
    return x + y + z;
}
```

```
int  maximize(int  x, int y, int z)
{
    int max;           //max la bien dia phuong
    max = x;
    if (max<y)         max = y;
    if (max<z)         max = z;
    return max;
}
```

1.1 HƯỚNG DẪN THỰC HÀNH BÀI 01

Trong bài thực hành này sinh viên tìm hiểu và thực hành cách sử dụng hàm

Bài toán:

Viết hàm tìm số lớn nhất trong 2 số. Áp dụng tìm số lớn nhất trong 4 số nhập vào từ bàn phím.

I. HƯỚNG DẪN BAN ĐẦU

Toán tử ?

- Cú pháp:

(biểu_thức_logic) ? Giá_trị_đúng : giá_trị_sai;

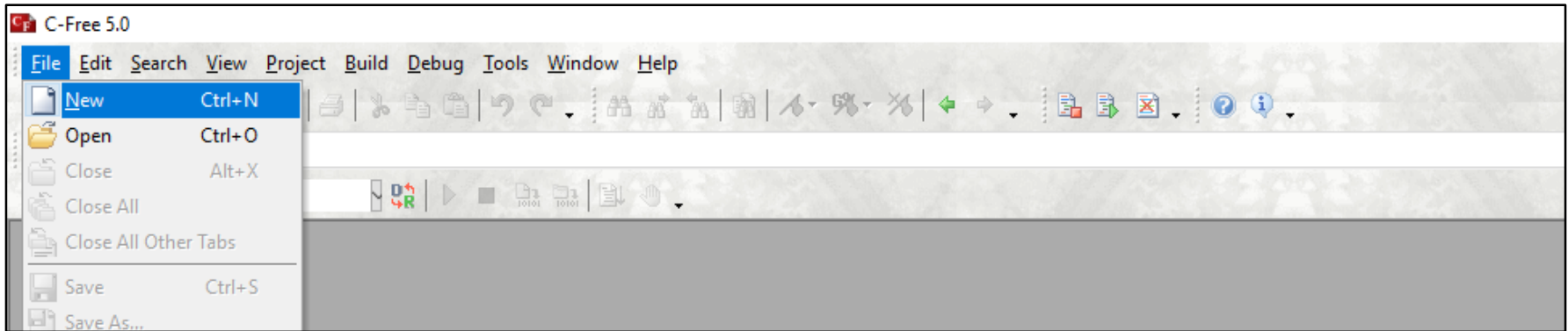
Ví dụ:

(x>y) ? x : y;

(x>0) ? 1 : 0;

1.1 HƯỚNG DẪN THỰC HÀNH BÀI 01

- **Bước 1:** Tạo mới một file*.cpp thực hiện thao tác File\New



- File mới xuất hiện, sinh viên chuyển sang bước 2 thực hiện gõ các câu lệnh theo các bước hướng dẫn.

1. 1 HƯỚNG DẪN THỰC HÀNH BÀI 01

Bước 2: Khai báo thư viện cần dùng

```
#include <iostream.h>
```

Bước 3: Định nghĩa hàm tìm số lớn nhất trong 2 số

```
int    max(int    x, int    y)
{
    if (x>y)
        return    x;
    else
        return    y;
}
```

Bước 4: Khai báo hàm main() là hàm chính của chương trình

```
void    main()
```

1. 1 HƯỚNG DẪN THỰC HÀNH BÀI 01

Bước 2: Khai báo thư viện cần dùng

```
#include <iostream.h>
```

Bước 3: Định nghĩa hàm tìm số lớn nhất trong 2 số

```
int    max(int    x, int    y)
{
    return (x>y) ? x:y;
}
```

Bước 4: Khai báo hàm main() là hàm chính của chương trình

```
void    main()
{
    int    i, n;    //i là biến điều khiển của vòng for
```

```
long int S=0;    //khai báo và khởi tạo biến tổng
```

1. 1 HƯỚNG DẪN THỰC HÀNH BÀI 01

Bước 5: Khai báo biến

```
int a, b, c, d;
```

Bước 6: Nhập dữ liệu

```
cout<<"Nhap vao 4 so nguyen:"<<endl;
```

```
cout<<"a = "; cin>>a;
```

```
cout<<"b = "; cin>>b;
```

```
cout<<"c = "; cin>>c;
```

```
cout<<"d = "; cin>>d;
```


1. 1 HƯỚNG DẪN THỰC HÀNH BÀI 01

Bước 7: Gọi hàm max ở trên để tìm số lớn nhất trong 4 số, in kết quả ra màn hình:

```
cout <<"So lon nhat la" <<max(max(a,  
b) , max(c, d) )<<endl;
```

1. 1 HƯỚNG DẪN THỰC HÀNH BÀI 01

```
#include <iostream.h>

int  max(int, int);           //khai báo nguyên mẫu

void main()

{
    int  a, b, c, d;

    cout<<"Nhap vao 4 so nguyen:"<<endl;

    cout<<"a = "; cin>>a;

    cout<<"b = "; cin>>b;

    cout<<"c = "; cin>>c;

    cout<<"d = "; cin>>d;
```

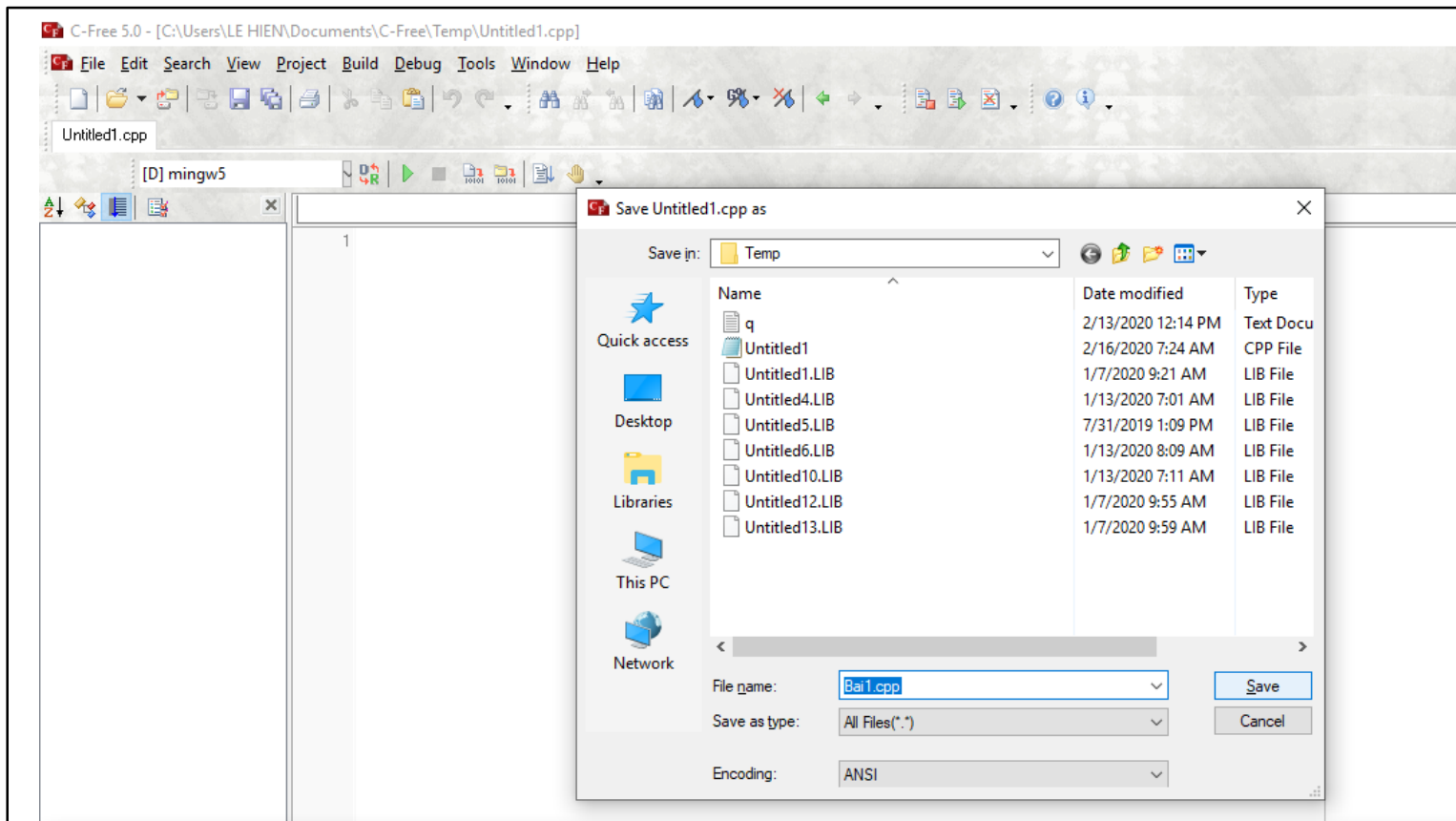
1. 1 HƯỚNG DẪN THỰC HÀNH BÀI 01

```
cout <<"So lon nhat trong 4 so la: ";  
cout <<max(max(a, b), max(c, d))<<endl;  
  
int    max(int    x, int    y)  
{  
    return (x>y) ? x:y;  
}
```

1.1 HƯỚNG DẪN THỰC HÀNH BÀI 01

- **Bước 8:** Lưu bài, chạy kiểm tra chương trình

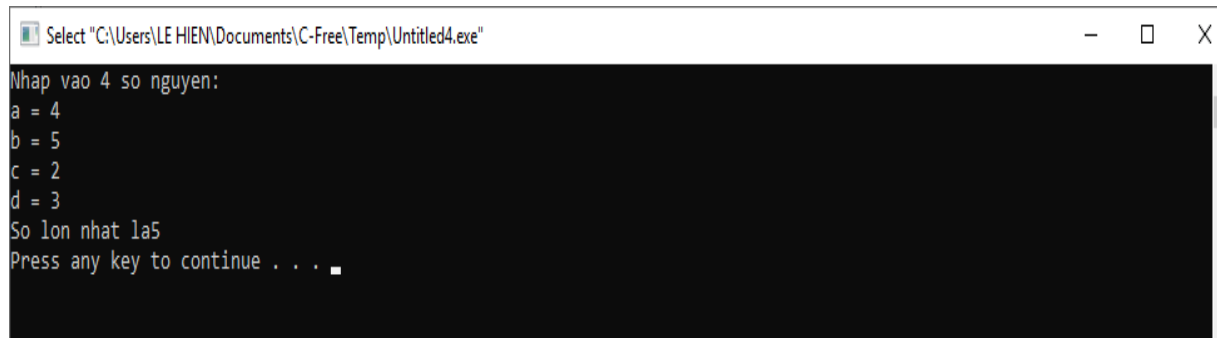
Sinh viên chọn trên thanh công cụ: File\Save\Gõ tên file cần lưu\Save



Ấn F5 để chạy và kiểm tra chương trình

1.1 HƯỚNG DẪN THỰC HÀNH BÀI 01

Lưu bài, biên dịch chương trình, nếu có lỗi phải sửa lỗi, nếu chương trình chạy đúng sẽ cho kết quả như sau:



```
Select "C:\\Users\\LE HIEN\\Documents\\C-Free\\Temp\\Untitled4.exe"
Nhap vao 4 so nguyen:
a = 4
b = 5
c = 2
d = 3
So lon nhat la 5
Press any key to continue . . .
```

Tóm lại: Trong bài thực hành 01 chúng ta đã biết cách xây dựng một chương trình cơ bản của C++, cách sử dụng hàm.

1.2 HƯỚNG DẪN THỰC HÀNH BÀI 02

Bài toán:

1. Xây dựng hàm kiểm tra một số có phải là số hoàn hảo không (Số hoàn hảo là một số có tổng các ước của nó (không kể nó) bằng chính nó. Ví dụ: Số 6 là số hoàn hảo vì tổng các ước $1+2+3=6$. Áp dụng tìm ra các số hoàn hảo trong phạm vi n .
2. Xây dựng hàm kiểm tra một số có phải là số nguyên tố không
Áp dụng tìm và in ra các số nguyên tố trong phạm vi n .

1.2 HƯỚNG DẪN THỰC HÀNH BÀI 02

Bước 1: Tạo file mới: tương tự như hướng dẫn ở bài thực hành 01

Bước 2: Khai báo thư viện cần dùng

```
#include <iostream.h>
```

1.2 HƯỚNG DẪN THỰC HÀNH BÀI 02

Bước 3: Khai báo hàm KTSHH

```
int    KTSHH(int    x)
{
    int tong = 0;
    for (int i=1; i<x; i++)
        if (x%i == 0)
            tong +=i;
    if (tong==x)
        return 1;
    return 0;
}
```


2.1 HƯỚNG DẪN THỰC HÀNH BÀI 02

```
int    main()
{
    int    n;
    cout<<"Nhap vao so nguyen n:";
    cin>>n;
    cout<<"Cac so hoan hao trong pham vi
    "<<n<<"la: "<<endl;
    for (int i=1;i<=n;i++)
        if (KTSHH(i)==1)
            cout<<i<<"\t";
    return 0;
}
```

1.2 HƯỚNG DẪN THỰC HÀNH BÀI 02

Bước 4: Khai báo hàm main() là hàm chính của chương trình

```
int main()
```

Bước 5: Khai báo hàm main() là hàm chính của chương trình

```
unsigned int n;
```

```
cout<<"Nhập vào số nguyên dương n: ";
```

```
cin>>n;
```

Bước 6: In kết quả ra màn hình:

```
cout<<"Các số hoán hạo trong phạm vi  
<<n<<" là:"<<endl;
```

```
for (int i = 1; i<=n; i++)
```

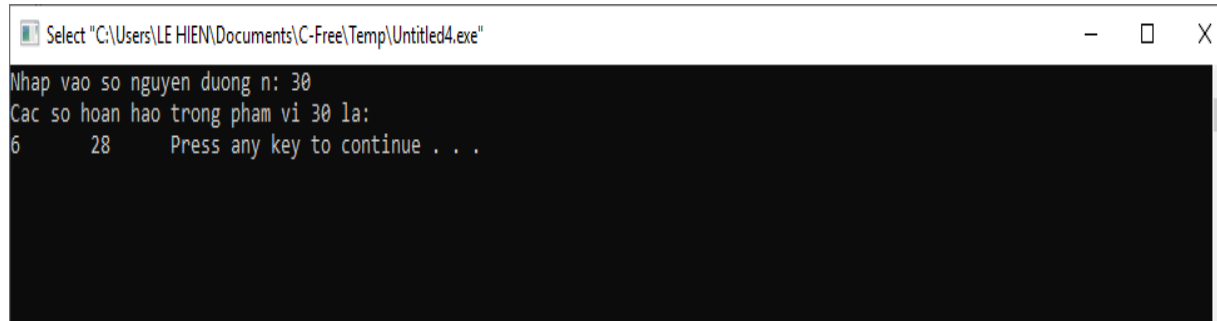
```
if (KTSHH(i) == 1)
```

```
cout<<i<<"\t";
```

1.2 HƯỚNG DẪN THỰC HÀNH BÀI 02

Bước 8: Thực hiện tương tự như bài thực hành 01

Biên dịch chương trình, nếu có lỗi phải sửa lỗi, nếu chương trình chạy đúng sẽ cho kết quả như sau:



```
Select "C:\\Users\\LE HIEN\\Documents\\C-Free\\Temp\\Untitled4.exe"
Nhap vao so nguyen duong n: 30
Cac so hoan hao trong pham vi 30 la:
6      28      Press any key to continue ...
```

- Chương trình hoàn chỉnh sinh viên xem trong tài liệu “B5_Tailieu_TTLTCB”

Tóm lại: Trong bài thực hành này chúng ta sẽ biết cách sử dụng câu lệnh do ... while.

II. HƯỚNG DẪN THƯỜNG XUYỀN

- Sinh viên tiếp tục thực hành các bài tập theo sự gợi ý hướng dẫn.

Trong phần này yêu cầu:

+ Một số phần trong bài sinh viên phải tự thực hiện (Ví dụ: tạo file ban đầu, khai báo thư viện, những hàm có tính chất tương tự ở ví dụ trước).

+ Sinh viên phải tự hoàn thiện chương trình và chạy đúng.

2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

Bài toán:

Viết hàm tìm UCLN của 2 số nguyên dương, áp dụng tìm UCLN của 4 số nguyên dương nhập vào từ bàn phím.

2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

Hàm tìm UCLN của 2 số

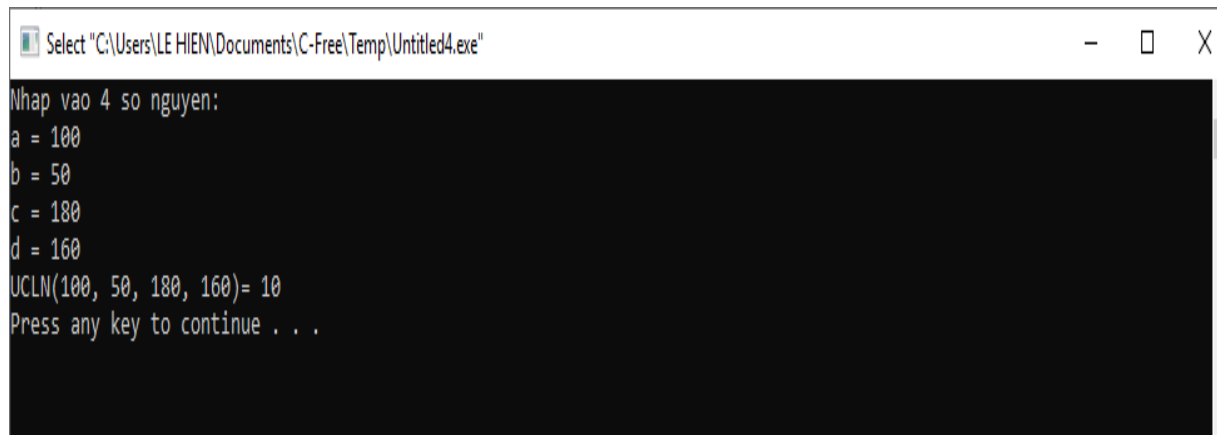
```
unsigned int    UCLN(unsigned int    m, unsigned
int    n)
{
    while (m != n)
        {
            if      (m>n)          m = m - n;
            else          n = n - m;
        }
    return  m;
}
```

2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

```
int    main()
{
    unsigned int    a, b, c, d;
    cout<<"Nhap vao 4 so nguyen:"<<endl;
    cout<<"a = "; cin>>a;
    cout<<"b = "; cin>>b;
    cout<<"c = "; cin>>c;
    cout<<"d = "; cin>>d;
    cout <<"UCLN(" <<a <<" , " <<b<<" , " <<c<<" ,
    " <<d<<" ) = " <<UCLN(UCLN(a, b), UCLN(c, d))<<endl;
    return 0;
}
```

2.1 HƯỚNG DẪN THỰC HÀNH BÀI 03

- Chương trình hoàn chỉnh sinh viên xem trong tài liệu “B5_Tailieu_TTLTCB”
- Kết quả khi chạy chương trình



```
Select "C:\Users\LE HIEN\Documents\C-Free\Temp\Untitled4.exe"
Nhap vao 4 so nguyen:
a = 100
b = 50
c = 180
d = 160
UCLN(100, 50, 180, 160)= 10
Press any key to continue . . .
```


2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

Bài toán:

Viết hàm tính $P(x,y)=x^y$. Áp dụng tính a^n với a và n nhập vào từ bàn phím.

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

Hàm lũy thừa

```
float luythua(int x, int y)
{
    float    p = 1;

    for (int i = 1; i<=y; i++)

        p = p*x;           //hoac    p    *= x;

    return    p;

}
```

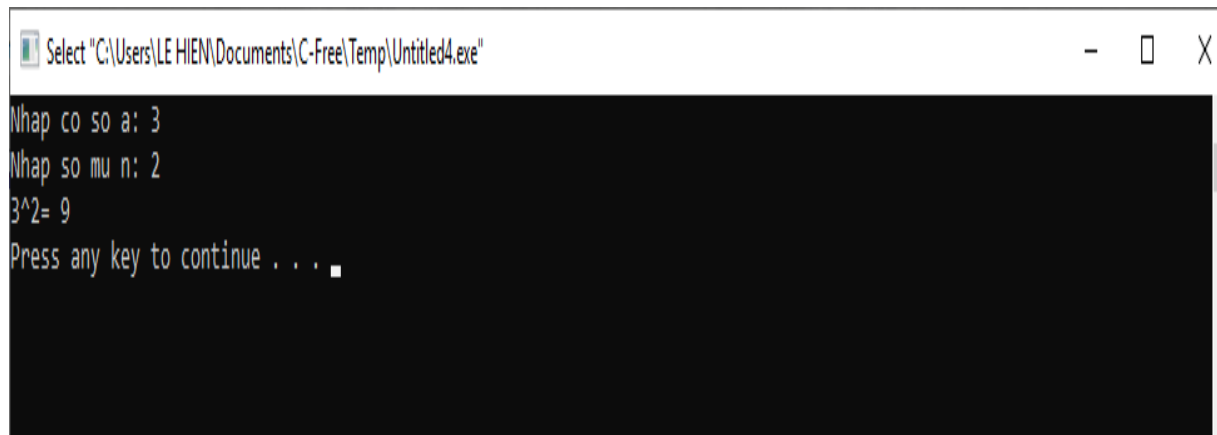
2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

Đoạn lệnh in ra bảng nhân n

```
int main ()
{
    int a, n;
    cout << "Nhap co so a: ";    cin >> a;
    cout << "Nhap so mu n: ";    cin >> n;
    cout<<a<<"^"<<n<<"= " << luythua(a,n) << endl;
    return 0;
}
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 04

- Chương trình hoàn chỉnh sinh viên xem trong tài liệu “B5_Tailieu_TTLTCB”
- Kết quả khi chạy chương trình



```
Select "C:\Users\LE HIEN\Documents\C-Free\Temp\Untitled4.exe"
Nhap co so a: 3
Nhap so mu n: 2
3^2= 9
Press any key to continue . . .
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 05

Bài toán:

Xây dựng hàm kiểm tra một số có phải là số nguyên tố không Áp dụng tìm và in ra các số nguyên tố trong phạm vi n.

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 05

Bước 3: Khai báo hàm kiểm tra nguyên tố

```
int      KTSNT (int      x)
{
    int count = 0;
    for (int i=1; i<=x; i++)
        if (x%i == 0)
            count++;
    if (count==2)
        return 1;
    return 0;
}
```

1.2 HƯỚNG DẪN THỰC HÀNH BÀI 05

```
int    main()
{
    unsigned    int    n;
    cout<<"Nhap vao so nguyen duong n: ";
    cin>>n;

    cout<<"Cac so nguyen to trong pham vi"<<n<<" la:"
<<endl;

    for (int    i = 1; i<=n; i++)
        if (KTSNT(i) == 1)
            cout<<i<<"\t";

    return 0;
}
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 06

Bài toán:

Viết hàm đệ quy tính giai thừa. Áp dụng tính tổ hợp chập k của n.

$$C_n^k = \frac{n!}{k! * (n-k)!}$$

$$n! = 1. 2. 3. 4 . \dots (n-1). n$$

$$n! = \begin{cases} 1 & \text{với } n = 0 \\ (n-1)! * n & \text{với } n \neq 0 \end{cases}$$

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 06

Định nghĩa hàm giai thừa

```
int    giai_thua (int    x)
{
    if (x==0)
        return 1;
    else
        return giai_thua (x-1) *x;
}
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 06

```
int    main ()
{
    unsigned int    n, k;

    float C;

    cout<<"Nhap n: ";    cin>>n;

    cout<<"Nhap k: ";    cin>>k;

    C=giai_thua (n) / (giai_thua (k) *giai_thua (n-k) ) ;

    cout<<"To hop chap "<<k<<" cua"<<n<<" la"
    <<C<<endl;

    return 0;
}
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 07

Bài toán:

1. Viết hàm tìm UCLN của 2 số nguyên dương m, n . Áp dụng tìm UCLN của 4 số nguyên dương a, b, c, d .

2. Viết hàm đệ quy tính

$$S = 1 + 2 + 3 + \dots + (n-1) + n.$$

3. Viết hàm đệ quy tìm UCLN của 2 số nguyên dương m, n . Áp dụng tìm UCLN của 3 số nguyên dương a, b, c .

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 07

Bài toán:

Viết hàm đệ quy tính

$$S = 1 + 2 + 3 + \dots + (n-1) + n.$$

$$S(n) = \begin{cases} 0 & \text{với } n = 0 \\ S(n-1) + n & \text{với } n > 0 \end{cases}$$

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 07

```
#include <iostream.h>

int    sum(int    x)
{
    if (x==0)
        return 0;
    else
        return sum(x-1) + x;
}

int    main()
{
    int    n;
    cout<<"Nhap n: ";    cin>>n;
    cout<<"S = "<<sum(n)<<endl;
    return 0;
}
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 08

Viết hàm đệ quy tìm UCLN của 2 số nguyên dương m, n . Áp dụng tìm UCLN của 3 số nguyên dương a, b, c .

$$\text{UCLN}(m, n) = \begin{cases} m & \text{với } m = n \\ \text{UCLN}(m-n, n) & \text{với } m > n \\ \text{UCLN}(m, n-m) & \text{với } m < n \end{cases}$$

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 08

```
#include <iostream.h>

int    UCLN(int    m, int n)
{
    if (m == n)
        return m;
    else
        if (m>n)
            return UCLN(m-n,n) ;
        else
            return UCLN(m,n-m) ;
}
```

2.2 HƯỚNG DẪN THỰC HÀNH BÀI 08

```
int    main ()
{
    int    a, b, c;

    cout<<"Nhap a: ";    cin>>a;
    cout<<"Nhap b: ";    cin>>b;
    cout<<"Nhap c: ";    cin>>c;

    cout<<"UCLN  ="<<UCLN (UCLN (a,b) ,  c)<<endl;

    return 0;
}
```


III. TỔNG KẾT

Kiến thức cần ghi nhớ trong buổi thực hành:

- 1 Khai báo và định nghĩa hàm
- 2 Lời gọi hàm
- 3 Viết chương trình sử dụng hàm
- 4 Phát hiện và sửa lỗi

Lời ngỏ

Trong quá trình học tập nếu sinh viên không hiểu phần nào thì liên hệ trao đổi với giảng viên qua hình thức gửi câu hỏi trên diễn đàn hoặc gửi vào email cho giảng viên.

III. GIAO NHIỆM VỤ TUẦN TIẾP THEO

1. Hoàn thành tất cả các bài thực hành trong buổi học và bài tập trong mục D file “B5_Baitap_TTLTCB.docx”.
2. Sinh viên đọc tài liệu và chuẩn bị trước nội dung học của tuần tiếp theo:
 - **Lập trình đệ quy**
 - + Hàm đệ quy
 - + Viết chương trình sử dụng hàm đệ quy
 - + Phát hiện và sửa lỗi.