

BÀI THỰC HÀNH SỐ 3 :: TOÁN HẠNG, TOÁN TỬ & BIỂU THỨC

MỤC TIÊU

1. Phân biệt và sử dụng được từ khóa, tên chuẩn, định danh, lời chú thích.
2. Biết tên, kích thước và phạm vi của các kiểu dữ liệu; ký tự (char), số nguyên (int, short, long, unsigned int, unsigned long), số thực (float, double, long double) và sử dụng đúng kiểu trong các bài toán cụ thể.
3. Hiểu, phân biệt và sử dụng được các đối tượng hằng và biến trong C.
4. Hiểu cách thực hiện các phép toán số học, lô gic, quan hệ trong C
5. Vận dụng các phép toán số học, lô gic, quan hệ một cách linh hoạt để giải quyết các bài toán tính toán đơn giản.
6. Sử dụng phép toán 3 ngôi (? :)
7. Sử dụng phép ép kiểu trong biểu thức
8. Sử dụng được các hàm cơ bản trong các thư viện chuẩn của C: Gọi hàm, truyền danh sách tham số và xác định giá trị trả về của hàm và/hoặc chức năng của hàm.

I. NỘI DUNG THỰC HÀNH: CÁC THÀNH PHẦN CƠ BẢN CỦA C

Vấn đề 1: Kích thước của kiểu dữ liệu cơ bản.

Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Size.CPP*

```
#include <stdio.h>
int main()
{
    printf("\nSize of char: %d\n", sizeof(char));
    printf("\nSize of int: %d\n", sizeof(int));
    printf("\nSize of short: %d\n", sizeof(short));
    printf("\nSize of float: %d\n", sizeof(float));
    return 0;
}
```

- 1.1. Dịch và chạy chương trình *Size.CPP* quan sát kết quả trên màn hình, ghi nhớ kích thước của mỗi kiểu dữ liệu.

STT	Kiểu dữ liệu	Kích thước (Byte)
1	char	
2	int	
3	short	
4	float	

- 1.2. Hàm *sizeof(x)* trả về một số nguyên cho biết kích thước của kiểu x, anh/chị hãy viết thêm các câu lệnh để hiển thị kích thước của các kiểu *long*, *unsigned int*, *unsigned long*, *double*, *long double* lên màn hình. Ghi nhớ kích thước của mỗi kiểu dữ liệu.

Vấn đề 2: Phạm vi lưu trữ của kiểu dữ liệu.

Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Range.CPP*. Chạy chương trình và phát hiện lỗi ngữ nghĩa. Giải thích lỗi và sửa lỗi để chương trình tính đúng trung bình cộng của 2 số nguyên a và b.

```
#include <stdio.h>
int main()
{
    short a, b, c, d;
    a = 30000;
    b = 30000;
    c = a+b;
    d=c/2;
    printf("\na=%d", a);
    printf("\nb=%d", b);
    printf("\nc=%d", c);
    printf("\nd=%d", d);
    return 0;
}
```

Vấn đề 3: Định danh (tên biến, hằng)

Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Note.CPP*

```
#include <stdio.h>
int main ()
{
    const float pi = 3.1416; //Định nghĩa hằng số pi
    float ban_kinh, chu_vi; //Khai báo biến ban_kinh và chu_vi
    /*
    Thuật toán tính chu vi đường tròn:
    1. Nhập dữ liệu: Bán kính
    2. Xử lý: Tính chu vi C = 2*r*pi
    3. Xuất thông tin: In kết quả lên màn hình
    */
    printf("\n-----CHUONG TRINH TINH CHU VI HINH TRON-----");
    printf("\n----- Nhập vào bán kính đường tròn:");
    scanf("%f", &ban_kinh);
    chu_vi = 2*ban_kinh*pi;
    printf("\n-----Chu vi của duong tron :%.4f", chu_vi);
    return 0;
}
```

- 3.1. Dịch và chạy chương trình *Note.CPP*, nhập vào bán kính đường tròn là một số không âm, kết thúc bằng ấn phím **enter**, quan sát kết quả trên màn hình.
- 3.2. Ở dòng 4, sửa tên biến **ban_kinh** bằng **ban-kinh**, và **chu_vi** bằng **chu vi**, dịch và chạy lại chương trình, giải thích lỗi và ghi nhớ về cách đặt tên trong ngôn ngữ C!

Vấn đề 4: Phạm vi hoạt động của biến

Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Avar.CPP*. Chạy chương trình và phát hiện lỗi cú pháp, giải thích lý do phát sinh lỗi và sửa code để chương trình chạy đúng.

```
#include <stdio.h>
int main()
{
    {    int a;    /* a có phạm vi toàn cục*/
        a = 1;
        {    int b; /* b chỉ có phạm vi trong khối lệnh*/
            b = 2;
            printf("\nGiá trị của a:%d\n", a);
            printf("\nGiá trị của b:%d\n", b);
        }
        printf("\nGiá trị của a:%d\n", a);
        printf("\nGiá trị của b:%d\n", b);
    }
    return 0;
}
```

Vấn đề 5: Địa chỉ và Giá trị của Biến

5.1. Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Addr.CPP*. Chạy chương trình và quan sát kết quả.

```
#include <stdio.h>
int main()
{
    char a='A';
    int b=10;
    float c;
    printf("\nNhập vào một số thực c ="); scanf("%f", &c);
    printf("\nGiá trị của a=%c. Địa chỉ của a=%p", a, &a);
    printf("\nGiá trị của b=%d. Địa chỉ của b=%p", b, &b);
    printf("\nGiá trị của c=%f. Địa chỉ của c=%p", c, &c);
    return 0;
}
```

5.2. Hãy bỏ ký tự **&** trước **&c** (ở dòng số 6 tính từ trên xuống), chạy chương trình và xem kết quả. Ghi nhớ điều này để tìm lời giải thích khi các anh/chị tìm hiểu sâu hơn về ngôn ngữ lập trình C ở các bài học tiếp theo!

5.3. Viết chương trình khai báo các biến có kiểu *long*, *unsigned int*, *unsigned long*, *double*, *long double* và gán giá trị hợp lệ. Hiển thị lên màn hình giá trị của biến và địa chỉ của các biến.

Ghi chú: Định dạng “địa chỉ” sử dụng **%p** và để lấy địa chỉ của một biến sử dụng toán tử **&**.

Vấn đề 6: Ký tự điều khiển

Bảng dưới đây cho biết các mã điều khiển, anh/chị hãy viết chương trình sử dụng các ký tự điều khiển (có mô tả cho trong bảng) để in ra màn hình văn bản được định dạng bên dưới.

\n	Xuống dòng
\r	Lùi về đầu dòng
\t	Ký tự tab, nhảy 1 đoạn có khoảng cách 0.5 inch
\v	Căn thẳng theo chiều dọc
\b	Backspace

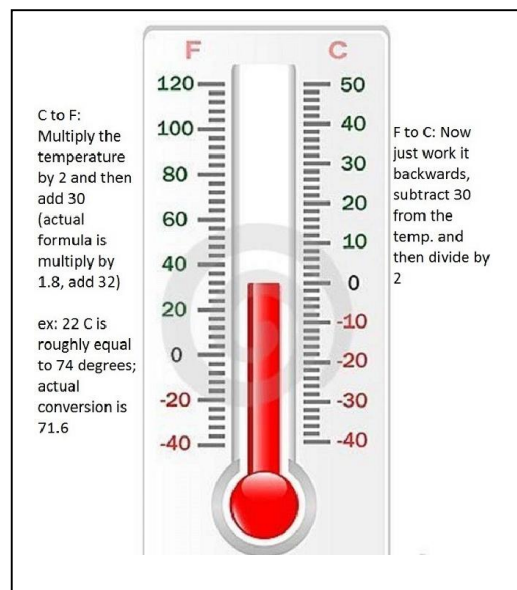
BANG DIEM: NHAP MON LAP TRINH

STT	HO DEM	TEN	DIEM TB
1	Nguyen Van	An	9.5
2	Nguyen Long	Son	8.2
3	Tran Nam	Trung	7.0

II. NỘI DUNG THỰC HÀNH: TOÁN TỬ & BIỂU THỨC**Vấn đề 7: Phép toán số học (+, -, *, /).**

Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Calc.CPP*

```
#include <stdio.h>
int main ()
{
    int do_F, do_C;
    printf("\n-----CHUONG          TRINH
    CHUYEN DOI NHIET DO-----");
    printf("\n-----Nhap vao nhiet do
    F:");
    scanf("%d", &do_F);
    printf("\n-----Nhiet do C tuong
    ung:%d", 5/9*(do_F-32));
    return 0;
}
```



- 7.1. Dịch và chạy chương trình *Calc.CPP*, nhập vào nhiệt độ F và quan sát kết quả trên màn hình để ghi vào bảng dữ liệu bên dưới:

Test	°F	°C
1	32	
2	73	
3	100	

- 7.2. Phát hiện lỗi ngữ nghĩa và sửa code để chương trình thực hiện đúng việc chuyển đổi độ F qua độ C, biết rằng: $^{\circ}\text{C} = \frac{5}{9}(^{\circ}\text{F}-32)$.

Vấn đề 8: Phép chia lấy dư - mod (%)

Để chuyển đổi giữa 2 đơn vị thời gian giây và phút, anh/chị có thể sử dụng kết hợp 2 phép toán chia (/) và mod (%). Cụ thể, để chuyển n giây thành phút và giây anh/chị thực hiện tuần tự các bước sau:

Bước 1: số_phút = n/60; //(phép chia lấy phần nguyên)

Bước 2: số_giây = n%60; //(phép chia lấy phần dư)

Bước 3: n giây = (số_phút)phút và (số_giây)giây;

8.1. Viết chương trình để chuyển n giây thành phút và giây, n là số nguyên dương nhập từ bàn phím, lưu vào **LAB03** với tên file *div_mod.CPP*

8.2. Phát triển chương trình *div_mod.CPP* để chuyển n giây thành **ngày, giờ, phút và giây**.

Vấn đề 9: Vận dụng mod (%)

Viết chương trình nhập vào số ngày ở khách sạn của một khách hàng sau đó thông báo lên màn hình số tiền mà họ phải trả (Lưu vào thư mục **LAB03** với tên file *Service.CPP*). Biết rằng công thức tính tiền của khách sạn như sau:

$$[\text{Tiền trả}] = \langle \text{số tuần} \rangle * 700000 + \langle \text{số ngày} \rangle * 120000.$$

Trong đó $\langle \text{số ngày} \rangle$ là những ngày dư ra, không nằm trong tuần. Ví dụ: Nếu số ngày lưu trú ở khách sạn là 12 ngày thì sẽ tương đương 1 tuần và 5 ngày.

Yêu cầu:

9.1. Nêu thuật toán

9.2. Xây dựng bảng dữ liệu mô tả các biến:

BIẾN	MÔ TẢ	GIÁ TRỊ HỢP LỆ	Kiểu

9.3. Viết chương trình

9.4. Xây dựng bộ dữ liệu test_case để chạy và kiểm thử chương trình.

Vấn đề 10: Phép toán quan hệ, lô gic và Toán tử 3 ngôi (? :)

Soạn thảo chương trình dưới đây trong editor của C-Free và lưu vào thư mục **LAB03** với tên file *Comp.CPP*

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    printf("\nNhập a ="); scanf("%d", &a);
    printf("\nNhập b ="); scanf("%d", &b);
    printf("\nNhập c ="); scanf("%d", &c);
    int dk = a<b && b<c;
    char kq = (dk==1)?'Y':'N';
    printf("\n%d, %d, %d sắp theo thu tu tang? %c\n",a,b,c, kq);
    return 0;}

```

10.1. Đọc hiểu chương trình *Comp.CPP*. Chạy chương trình với các bộ dữ liệu vào (*input*) bên dưới, quan sát kết quả trên màn hình và ghi vào cột *output* của bảng:

Test	input			output
	a	b	c	Sắp tăng?
1	3	4	5	
2	3	3	5	
3	5	4	3	

10.2. Phát triển chương trình *Comp.CPP* để kiểm tra các điều kiện dưới đây đúng hay sai?

1. Không giảm: $a \leq b \leq c$
2. Không tăng: $a \geq b \geq c$
3. Sắp giảm: $a > b > c$

III. BÀI TẬP

Bài 0015

Viết chương trình nhập vào hai cạnh vuông của một tam giác vuông. Sau đó, tính diện tích và cạnh huyền của tam giác này.

Bộ dữ liệu mẫu:

Input		Output	
Cạnh vuông 1 (a)	Cạnh vuông 2 (b)	Diện tích (S)	Cạnh huyền (c)
3	4	6	5
5	5.25	13.125	7.25
1	2	1	2.236068

Hướng dẫn:

- Công thức tính diện tích tam giác vuông: $S = \frac{a*b}{2}$
- Cạnh huyền $c = \sqrt{a^2 + b^2}$, hàm tính căn bậc 2 của a là `sqrt(a)` trong thư viện `<math.h>`

Bài 0016

Viết chương trình nhập 2 số nguyên vào 2 biến a và b. Sau đó hoán đổi giá trị của 2 biến a và b

Bộ dữ liệu mẫu:

Input		Output	
a	b	a	b
1	2	2	1
3	5	5	3
3	3	3	3

Ghi chú: Có thể thực hiện hoán đổi giá trị của a và b mà không cần đến biến trung gian bằng cách thực hiện tuần tự các câu lệnh sau:

1. $a = a + b$
2. $b = a - b$
3. $a = a - b$

Hãy viết chương trình thực hiện công việc theo cách trên.

Bài 0017

Nhập 2 số nguyên dương a và b có đủ 3 chữ số. Trình bày phép toán nhân a với b ra màn hình

Bộ dữ liệu mẫu:

Input		Output
<i>a</i>	<i>b</i>	<i>Màn hình</i>
123	456	<pre> 123 x 456 ----- 738 615 492 ----- 56088 </pre>
162	802	<pre> 162 x 802 ----- 324 0 1296 ----- 129924 </pre>

Hướng dẫn:

- Vận dụng toán tử / (chia lấy nguyên) và % (chia lấy dư) để thực hiện thao tác nhân tuần tự.
- Xem lại cú pháp của hàm **printf** trong thư viện **<stdio.h>** để trình bày kết quả lên màn hình theo yêu cầu.
- Chú ý kích thước kiểu dữ liệu.

Bài 0018

Nhập 1 số nguyên x. Xuất ra màn hình kết quả của biểu thức sau:

$$1 + \frac{x^2}{1 + \frac{x^2}{1 + x^2}}$$

Bộ dữ liệu mẫu:

Input	Output
<i>x</i>	<i>Kết quả biểu thức</i>
1	0.6
3	1.5689
5	1.8188
0	0

Bài 0019

Nhập 2 số nguyên x và y. Xuất ra màn hình kết quả của biểu thức sau:

$$\frac{\sqrt{1 + x^2 + y^2}}{3 + (x + y)^2}$$

Bộ dữ liệu mẫu:

Input		Output
x	y	Kết quả biểu thức
1	0	0.3536
0	1	0.3536
2	3	0.1336

Hướng dẫn: Vận dụng các hàm toán học được xây dựng sẵn trong thư viện <math.h>

Bài 0020

1) Khi nào thì bộ 3 số thực a, b và c là độ dài các cạnh của một tam giác vuông? Viết biểu thức điều kiện dưới dạng: **tam_giac_vuong = <biểu thức điều kiện>;**

2) Xây dựng bảng dữ liệu mô tả các biến:

BIẾN	MÔ TẢ	GIÁ TRỊ HỢP LỆ	KIỂU

3) Viết chương trình nhập vào 3 số a, b, c và thông báo lên màn hình ‘Y’ nếu chúng là bộ 3 cạnh của tam giác vuông hoặc ‘N’ nếu chúng không phải là các cạnh của một tam giác vuông.

English

Q1: There is a four-digit number N1N2N3C that is used for Customer Account Number (CAN). The right most digit “C” can be calculated as follows:

$$C = (N1*3 + N2*5 + N3*7) \bmod 10$$

Which of the following is a correct CAN? Here, x mod y returns the remainder when x is divided by y.

a) 7714

b) 7715

c) 9690

d) 9695

Q2: An integer mis stored in a register as a binary value. If this value is shifted to the left by three bits and mis added to the shifted value, how many times as large as mis the resulting number? Here, no overflow occurs.

a) 4

b) 7

c) 8

d) 9

Q3: There are a total of 1,900 students in a school, 553 of whom are taking a course in computer science, 667 of whom are in mathematics, and 290 of whom are in both computer science and mathematics. How many students are not taking a course in either computer science or in mathematics?

a) 680

b) 930

c) 970

d) 1260