

a. Bài tập cơ bản (kiểu dữ liệu, biểu thức...)

1. Nhận một giá trị góc (số nguyên) từ bàn phím. Sau đó hiện thông báo góc đã nhập thuộc góc phần tư thứ mấy.

Gợi ý: $(\text{goc}/90)\%4 + 1$

2. Nhận từ bàn phím một số có ba chữ số. In ra chữ số phần trăm, chục, đơn vị của số đó.
3. Lập chương trình thực hiện các công việc sau:
 - a. Nhập 3 số **x, y, z** bất kì.
 - b. Tính giá trị biểu thức **F**:

$$F = \frac{x + y + z}{x^2 + y^2 + 1} - |x - z \cos(y)|$$

- c. Đưa giá trị tính được của **F** ra màn hình dưới dạng:

Giá trị của F = giá trị

4. Lập chương trình thực hiện các công việc sau:
 - a. Nhập 3 số **a, b, c** bất kì
 - b. Kiểm tra xem **a, b, c** có thoả mãn điều kiện là 3 cạnh của 1 tam giác. Nếu có thì:
 - c. Tính chu vi **M**, diện tích **S** của tam giác theo 3 cạnh **a, b, c**.
 - d. Đưa các giá trị **S, M** tính được ra màn hình:

Chu vi của tam giác M = giá trị

Diện tích của tam giác S = giá trị

5. Viết chương trình tính $S = x^y$ (**x, y** là số thực)

b. Cấu trúc if/if...else và switch

1. Nhập vào hai số nguyên a, b. In ra màn hình giá trị lớn nhất.
2. Cho ba số a, b, c đọc vào từ bàn phím. Hãy tìm giá trị lớn nhất của ba số trên và in ra kết quả.
3. Cho ba số a, b, c đọc vào từ bàn phím. Hãy in ra màn hình theo thứ tự tăng dần các số. (Chỉ được dùng thêm hai biến phụ).
4. Viết chương trình nhập vào một số nguyên n gồm ba chữ số. Xuất ra màn hình chữ số lớn nhất ở vị trí nào?
Ví dụ: n=291. Chữ số lớn nhất nằm ở hàng chục (9).
5. Viết chương trình nhập vào số nguyên n gồm ba chữ số. Xuất ra màn hình theo thứ tự tăng dần của các chữ số.
Ví dụ: n=291. Xuất ra 129.
6. Nhập vào ngày, tháng, năm. Kiểm tra xem ngày, tháng, năm đó có hợp lệ hay không? In kết quả ra màn hình.
7. Nhập vào giờ, phút, giây. Kiểm tra xem giờ, phút, giây đó có hợp lệ hay không? In kết quả ra màn hình.
8. Viết chương trình nhập vào ngày, tháng, năm hợp lệ. Cho biết năm này có phải là năm nhuận hay không? In kết quả ra màn hình.
9. Viết chương trình tính diện tích và chu vi các hình: tam giác, hình vuông, hình chữ nhật và hình tròn với những thông tin cần được nhập từ bàn phím.
10. Viết chương trình tính tiền cước TAXI. Biết rằng:
 - KM đầu tiên là 5000đ.
 - 200m tiếp theo là 1000đ.
 - Nếu lớn hơn 30km thì mỗi km thêm sẽ là 3000đ.Hãy nhập số km sau đó in ra số tiền phải trả.
11. Nhập vào 3 số nguyên dương a, b, c. Kiểm tra xem 3 số đó có lập thành tam giác không? Nếu có hãy cho biết tam giác đó thuộc loại nào? (Cân, vuông, đều, ...).
12. Viết chương trình nhập vào số nguyên dương n. Kiểm tra xem n có phải là số chính phương hay không? (số chính phương là số khi lấy căn bậc 2 có kết quả là nguyên).
13. Giải phương trình bậc hai $AX^2 + BX + C = 0$ (A, B, C là các số thực)

14. Giải hệ phương trình bậc nhất

$$A_1X + B_1Y = C_1$$

$$A_2X + B_2Y = C_2$$

Gợi ý: sử dụng kiến thức ở môn Đại số tuyến tính – tính định thức

c. Cấu trúc lặp

1. Viết chương trình in ra màn hình hình chữ nhật đặc kích thước $m \times n$ (m, n nhập từ bàn phím).

Ví dụ: Nhập $m=5, n=4$

* * * * *

* * * * *

* * * * *

* * * * *

2. Viết chương trình in ra màn hình hình chữ nhật rỗng kích thước $m \times n$ (m, n nhập từ bàn phím).

Ví dụ: Nhập $m=5, n=4$

* * * * *

* * * * *

* * * * *

* * * * *

3. Viết chương trình in ra màn hình tam giác vuông cân đặc có độ cao h (h nhập từ bàn phím).

Ví dụ: Nhập $h=4$

*

* *

* * *

* * * *

4. Viết chương trình in ra màn hình tam giác cân rỗng có độ cao h (h nhập từ bàn phím).

Ví dụ: Nhập $h=4$

*

* *

* * *

* * * *

5. Viết chương trình in ra màn hình tam giác cân đặc có độ cao h (h nhập từ bàn phím).

Ví dụ: Nhập h=4

```
      *
     ***
    *****
   *********
  ***********
```

6. Viết chương trình in ra màn hình tam giác cân rỗng có độ cao h (h nhập từ bàn phím).

Ví dụ: Nhập h=4

```
      *
     *  *
    *    *
   *      *
  *        *
 *          *
*            *
```

7. Viết chương trình nhập số nguyên dương n. Liệt kê n số nguyên tố đầu tiên.

8. Viết chương trình nhập vào hai số nguyên dương a và b. Tìm ước số chung lớn nhất và bội số chung nhỏ nhất của a và b.

9. Viết chương trình đếm số ước số của số nguyên dương N.

Ví dụ: N=12

số ước số của 12 là 6

10. Một số hoàn thiện là một số có tổng các ước số của nó (không kể nó) bằng chính nó. Hãy liệt kê các số hoàn thiện nhỏ hơn 5000.

Ví dụ: số 6 là số hoàn thiện vì tổng các ước số là $1+2+3=6$

11. Nhập vào ngày, tháng, năm. Cho biết đó là ngày thứ mấy trong năm.

12. Viết chương trình nhập 1 số nguyên n, rồi tính và in ra phần tử thứ n của dãy số Fibonacci biết rằng :

$$F_1 = F_2 = 1$$

$$F_n = F_{n-1} + F_{n-2} \text{ với } n > 2$$

13. Viết chương trình in bảng cửu chương từ 2 đến 9

14. Viết chương trình tính n! (đưa ra 2 lời giải là dùng for và while)

15. Dùng lệnh **while** tính tổng các số nguyên từ 1 đến 10.

16. Hãy đếm lượng chữ số của số nguyên dương n

17. Hãy sử dụng vòng lặp for để xuất các ký tự từ A tới Z

d. Bài tập tổng hợp

1. Lập chương trình thực hiện các công việc sau:

Nhập số nguyên $n < 10$ bất kỳ và in ra tam giác pascal sao cho cân đối ,

Ví dụ $n=5$

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
```

2. Một tam giác vuông có thể có tất cả các cạnh là các số nguyên. Tập của ba số nguyên của các cạnh của một tam giác vuông được gọi là bộ ba Pitago. Đó là tổng bình phương của hai cạnh bằng bình phương của cạnh huyền, chẳng hạn bộ ba Pitago (3, 4, 5). Viết chương trình tìm tất cả các bộ ba Pitago như thế sao cho tất cả các cạnh không quá 500.

3. Lập chương trình thực hiện các công việc sau:

Nhập số thực x và sai số ϵ (kiểu float) rồi tính S theo công thức sau (với sai số $= \epsilon$ có giá trị bằng 1 phần ngàn, 1 phần triệu) :

$$\sin(x) \approx S = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!}$$

Sau đó in ra S với 12 chữ số thập phân và in ra $\sin(x)$ (theo hàm tính sin của thư viện `cmath`), đồng thời so sánh kết quả . Sau đó sửa lại kiểu của các biến thành `double` và so sánh lại kết quả !.

4. Lập chương trình thực hiện các công việc sau:

a. Nhập số **epsilon** <1 từ bàn phím

b. Tính số **e** theo công thức:

$$e = 1 + 1/(1!) + 1/(2!) + \dots + 1/(n!)$$

quá trình tính dừng khi $1/(n!) < \epsilon$

c. Đưa kết quả ra màn hình

5. Lập chương trình thực hiện các công việc sau:

Dùng vòng **while** hoặc **do...while** để tính số **pi** theo công thức:

$$\pi/4 = 1 - 1/3 + 1/5 - 1/7 + \dots + (-1)^i \cdot 1/(2i+1)$$

với độ chính xác **epsilon** nhập từ bàn phím. Khi đó ta sẽ có số **pi** với sai số **epsilon** khi **i** đủ lớn để:

$$1/(2i+1) < \text{epsilon}$$

6. Lập chương trình thực hiện các công việc sau:

Dùng vòng **while** hoặc **do...while** để tính số thực **f** theo công thức :

$$f = 1 - 3x + 5x^3/4! - 7x^6/7! + 9x^9/10! \dots$$

với độ chính xác **epsilon** nhập từ bàn phím

7. Lập chương trình thực hiện các công việc sau:

Nhập số nguyên **n < 20** bất kỳ và 1 ký tự **ch** bất kỳ :

- In ra hình vuông gồm **n** dòng, mỗi dòng có **n** ký tự **ch** . (lưu ý, in thế nào để được hình tương đối vuông, không quá lệch)
- In ra lần lượt 4 nửa hình vuông chia bởi 2 đường chéo chính
- Kết hợp 4 nửa hình vuông kể trên để in ra hình 1 cái quạt có 4 cánh, rồi tạo hiệu ứng cánh quạt xoay trên màn hình. (lưu ý, trên devC, để xóa màn hình ta dùng lệnh `system("cls")`);
- Ví dụ **n=5, ch = A**

```
A A A A A      A
  A A A A      A A
    A A A    A A A
      A A  A A A A
        A A A A A A A A
          A A A A  A A
            A A A    A A A
              A A      A A A A
                A        A A A A A
```

米	米	米	米	米				米
		米	米	米	米		米	米
			米	米	米		米	米
				米	米	米	米	米
米	米	米	米	米	米	米	米	米
米	米	米	米	米	米			
米	米	米			米	米	米	
米	米				米	米	米	米
米					米	米	米	米