

# BÀI TẬP THỰC HÀNH KỸ THUẬT LẬP TRÌNH

---

## PHẦN 2. VIẾT CHƯƠNG TRÌNH

**Bài 1.** Viết chương trình giải phương trình bậc 2.

**Bài 2.** Viết chương trình nhập vào 3 số a, b, c. Kiểm tra 3 số được nhập vào có phải là 3 cạnh tam giác? Nếu đúng là 3 cạnh tam giác, cho biết loại tam giác?

**Bài 3.** Viết chương trình tính tổng các số chẵn từ 2 đến N (nguyên dương, nhập từ bàn phím).

**Bài 4.** Viết chương trình kiểm tra số nguyên dương N nhập vào từ bàn phím có phải là số nguyên tố không?

**Bài 5.** Viết chương trình sắp xếp mảng các số nguyên theo chiều tăng dần giá trị các phần tử.

**Bài 6.** Viết chương trình tính  $S = 4! + 5! + 7! + 9!$

**Bài 7.** Viết chương trình tìm ước số chung lớn nhất của hai số nguyên nhập vào từ bàn phím.

**Bài 8.** Viết chương trình tìm bội số chung nhỏ nhất của hai số nguyên nhập vào từ bàn phím.

**Bài 9.** Viết chương trình tính tổng các số lẻ từ 1 đến n.

**Bài 10.** Viết chương trình sắp xếp mảng các số nguyên theo chiều giảm dần giá trị các phần tử.

**Bài 11.** Viết chương trình tìm giá trị nhỏ nhất của mảng các số nguyên.

**Bài 12.** Viết chương trình tìm giá trị lớn nhất của mảng các số nguyên.

**Bài 13.** Viết chương trình tìm phần tử cuối cùng của mảng chứa giá trị nhỏ nhất.

**Bài 14.** Viết chương trình có định nghĩa hàm tìm giá trị lớn nhất trong 3 số nguyên cho trước.

**Bài 15.** Viết chương trình kiểm tra một chuỗi là chuỗi số nguyên (chuỗi chứa tất cả ký tự số, trong đó ký tự đầu tiên khác không hoặc dấu âm).

**Bài 16.** Viết chương trình chuyển đổi chuỗi số nguyên sang số nguyên.

**Bài 17.** Viết chương trình cho phép nhập vào một chuỗi và in ra màn hình chuỗi đảo ngược.

**Bài 18.** Viết chương trình phân tích một số nguyên thành các thừa số nguyên tố. Ví dụ: Số 28 được phân tích thành  $2 \times 2 \times 7$ .

**Bài 19.** Viết chương trình liệt kê tất cả các số nguyên tố nhỏ hơn n cho trước.

**Bài 20.** Viết chương trình liệt kê n số nguyên tố đầu tiên.

**Bài 21.** Dãy số Fibonacci được định nghĩa như sau:  $F_0=1, F_1=1, F_n=F_{n-1} + F_{n-2}$  với  $n \geq 2$ . Hãy viết chương trình tìm số Fibonacci thứ  $k$ .

**Bài 22.** Viết chương trình liệt kê tất cả các xâu nhị phân độ dài  $n$ .

**Bài 23.** Viết chương trình liệt kê tất cả các tập con  $k$  phần tử của  $1, 2, \dots, n$  ( $k \leq n$ ).

**Bài 24.** Viết chương trình liệt kê tất cả các hoán vị của  $1, 2, \dots, n$ .

**Bài 25.** Viết chương trình liệt kê tất cả các số nguyên tố có 5 chữ số sao cho tổng của các chữ số trong mỗi số nguyên tố đều bằng  $S$  cho trước.

**Bài 26.** Nhập một số nguyên dương  $n$  từ bàn phím. Hãy liệt kê các số Fibonacci nhỏ hơn  $n$  là số nguyên tố.

**Bài 27.** Số hoàn hảo là số bằng tổng các ước thực sự của nó. Ví dụ:  $6 = 1+2+3$ . Viết chương trình kiểm tra số nguyên dương  $n$  nhập vào từ bàn phím có phải là số hoàn hảo không.

**Bài 28.** Viết chương trình thực hiện nhập một xâu ký tự. Tìm từ dài nhất trong xâu và vị trí. (Nếu có nhiều từ cùng độ dài thì chọn từ đầu tiên tìm thấy).

**Bài 29.** Viết chương trình thực hiện nhập một xâu theo cấu trúc: *họ ... tên đệm .... tên*. Chuyển xâu đó thành dạng: *tên ... họ ... tên đệm* và hiển thị kết quả.

**Bài 30.** Số đối xứng (Palindrome) là số nhận được khi thực hiện đảo ngược các chữ số của nó cho kết quả bằng số ban đầu. Ví dụ: 121, 3553. Viết chương trình kiểm tra số nguyên dương  $n$  nhập vào từ bàn phím có phải là số đối xứng (Palindrome) không. Nếu đúng thì hiển thị thông báo “Số vừa nhập là số đối xứng”, ngược lại hiển thị thông báo “Số vừa nhập không phải là số đối xứng”.

**Bài 31.** Viết chương trình kiểm tra xâu nhập vào từ bàn phím có phải là Palindrome không. Chuỗi ký tự được gọi là Palindrome khi thực hiện đảo ngược các ký tự của chuỗi thì nhận được kết quả là chuỗi ban đầu. Ví dụ: “Madam” là Palindrome. (Không phân biệt chữ hoa chữ thường).

**Bài 32.** Nhập một xâu ký tự. Đếm số từ của xâu ký tự đó. Ví dụ “ Lớp học ” có 2 từ.

**Bài 33.** Viết chương trình thực hiện các thao tác sau đối với một chuỗi cho trước

- Đếm số lượng từ ‘Java’ có trong chuỗi.
- Lấy vị trí của từ ‘Java’ cuối cùng có trong chuỗi.
- Thay thế từ ‘Java’ bằng từ ‘java’ trong chuỗi.
- Đếm số lượng ký tự là số có trong chuỗi.

**Bài 34.** Viết chương trình thực hiện các thao tác sau đối với chuỗi cho trước.

- Đếm số lượng ký tự ‘a’ có trong chuỗi.
- Kiểm tra chuỗi có chứa từ ‘Java’.
- Kiểm tra chuỗi có bắt đầu bằng từ ‘Write’.
- Kiểm tra chuỗi có kết thúc bằng từ ‘easily’.

**Bài 35.** Viết chương trình nhập vào ma trận A có n dòng, m cột, các phần tử là những số nguyên lớn hơn 0 và nhỏ hơn 100 được nhập từ bàn phím. Thực hiện các yêu cầu sau:

- a. Tìm phần tử lớn nhất cùng với chỉ số.
- b. Tìm và in ra các phần tử là số đối xứng (Palindrome).
- c. Tìm và in ra các phần tử là số nguyên tố (các phần tử không phải là số nguyên tố được thay bằng số 0)
- d. Sắp xếp tất cả các cột theo thứ tự tăng dần và hiển thị kết quả.

**Bài 36.** Viết chương trình liệt kê các số nguyên có từ 5 đến 7 chữ số thỏa yêu cầu:

- a. Là số nguyên tố.
- b. Là số đối xứng (palindrome).

**Bài 37.** Viết chương trình liệt kê các số nguyên có 7 chữ số thỏa yêu cầu sau:

- a. Tổng các chữ số của số đó là số đối xứng (palindrome).
- b. Mỗi chữ số đều là số nguyên tố.

**Bài 38.** Viết chương trình nhập vào mảng A gồm n phần tử, các phần tử là những số nguyên lớn hơn 0 và nhỏ hơn 100 được nhập từ bàn phím. Thực hiện các yêu cầu sau:

- a. Tìm phần tử lớn thứ 2 cùng với chỉ số.
- b. Sắp xếp mảng theo thứ tự giảm dần.
- c. Nhập số nguyên x và chèn x vào mảng A sao cho vẫn đảm bảo thứ tự sắp xếp giảm dần của mảng A.

---HẾT---