

Kí tự và xâu kí tự

Một chương trình C được tạo thành bởi sự kết hợp giữa các kí tự với nhau theo một trật tự và nguyên tắc nào đó. Đây là điều quen thuộc với chúng ta ngay từ những bài học đầu tiên, tuy vậy có thể chúng ta chưa nhận ra.

Nội dung của chương này bao gồm:

1. Cơ bản về kí tự và xâu kí tự
2. Giới thiệu về các hàm xử lý kí tự đơn
3. Giới thiệu về các hàm chuyển đổi xâu kí tự
4. Hàm thư viện chuẩn vào ra
5. Các hàm xử lý xâu kí tự
6. Các hàm so sánh xâu kí tự
7. Các hàm tìm kiếm xâu kí tự
8. Hàm liên quan tới bộ nhớ của thư viện xử lý chuỗi
9. Một số hàm khác được xây dựng trong thư viện chương trình

1. Cơ bản về kí tự và xâu kí tự

Kí tự

- Kí tự có thể là các chữ cái a, b, c, X, các chữ số 0, 1, 2, các kí tự khác: +, -, *, ?.
- Kí tự trong ngôn ngữ C là các kí tự hằng và thường được đại diện bởi một số nguyên lưu trong kiểu char. Ví dụ kí tự 'a' có giá trị nguyên tương ứng là 97; kí tự 'A' có giá trị nguyên tương ứng là 65.
- Xem bảng mã ASCII để biết thêm chi tiết.
- Ví dụ về một biến chứa kí tự: `char c = 'A';`

Xâu kí tự

- Xâu kí tự là một loạt các kí tự được coi như một thành phần thống nhất.
- Xâu kí tự có thể là sự kết hợp giữa các chữ cái, các chữ số, các kí tự đặc biệt.
- Xâu kí tự hằng thường để trong nháy kép, ví dụ:

- “Hello World!” là một chuỗi ký tự được ghép từ các ký tự: ‘H’ ‘e’ ‘l’ ‘l’ ‘o’ ‘W’ ‘o’ ‘r’ ‘l’ ‘d’ ‘!’.
- Ta dùng mảng char để lưu chuỗi ký tự:
 - `char hello[] = “Hello World!”;`
 - `const char *phoneNumber = “0978556600”;`
- Ký tự cuối cùng trong chuỗi ký tự luôn là ký tự `NULL(‘\0’)`. Do đó một mảng char chứa từ “Hello” sẽ bao gồm: {‘H’, ‘e’, ‘l’, ‘l’, ‘o’, ‘\n’}. Như vậy ta cần mảng char có 6 phần tử để lưu chuỗi “Hello”. Mở rộng ra: chuỗi có độ dài n thì cần mảng char có n+1 phần tử để lưu trữ.
- Vậy, chuỗi hello trên tương đương với:
 - `char hello[] = {‘H’, ‘e’, ‘l’, ‘l’, ‘o’, ‘W’, ‘o’, ‘r’, ‘l’, ‘d’, ‘!’, ‘\0’};`

Kết Luận: chuỗi ký tự là mảng của các ký tự, do đó có tính chất của mảng, con trỏ, luôn kết thúc bằng ký tự ‘\0’. Kích thước mảng để lưu trữ chuỗi ký tự tối thiểu phải là $n + 1$. Trong đó n là số lượng ký tự cần lưu trữ.

2. Giới thiệu các hàm xử lý ký tự đơn:

Trong thư viện `<ctype.h>` có sẵn một số hàm sử dụng tham số là giá trị nguyên không dấu hoặc EOF để kiểm tra, thao tác với các dữ liệu ký tự. Bao gồm:

- `int isblank(int c);` Trả về true (1) nếu ký tự trong tham số là ký tự khoảng trắng và false (0) trong các trường hợp còn lại.
- `int isdigit(int c);` Trả về true nếu ký tự trong tham số là ký tự chữ số từ ‘0’ tới ‘9’ và trả về false trong các trường hợp còn lại.
- `int isalpha(int c);` Trả về true nếu giá trị trong tham số là một ký tự chữ cái, trả về false trong các trường hợp còn lại.
- `int isalnum(int c);` Trả về true nếu ký tự trong tham số là ký tự chữ cái hoặc chữ số, false trong các trường hợp còn lại.

- `int isxdigit(int c);` Trả về true nếu kí tự trong tham số là kí tự đại diện của biểu diễn cơ số 16(0->9, A,B,C,D,E,F). Trả về false trong các trường hợp còn lại.
- `int islower(int c);` Trả về true nếu tham số là kí tự thường. Trả về false trong các trường hợp còn lại.
- `int isupper(int c);` Trả về true nếu tham số là kí tự hoa. Trả về false trong các trường hợp còn lại.
- `int toupper(int c);` Biến kí tự thường thành kí tự hoa.
- `int tolower(int c);` Biến kí tự hoa thành kí tự thường.
- `int isspace(int c);` Trả về true nếu tham số là kí tự khoảng trắng hoặc '\n', '\v', '\t', '\r', '\f'. Trả về false trong các trường hợp còn lại.
- `int iscntrl(int c);` Trả về true nếu tham số là kí tự điều khiển. Trả về false trong các trường hợp còn lại.
- `int ispunct(int c);` Trả về true nếu tham số là kí tự khác khoảng trắng, chữ cái, chữ số. Trả về false trong các trường hợp còn lại.