

CHUỖI KÝ TỰ



1. Chuỗi ký tự
2. Các tác vụ trên chuỗi ký tự

1. Chuỗi ký tự



1.1. Khái niệm

1.2. Khai báo, khởi tạo

1.3. Nhập xuất chuỗi

1.4. Một số hàm thông dụng trong thư viện

1.1. Khái niệm chuỗi ký tự



- Kiểu char chỉ chứa được một ký tự. Để lưu trữ một chuỗi (nhiều ký tự) ta sử dụng mảng (một chiều) các ký tự.
- Chuỗi ký tự kết thúc bằng ký tự “\0” (null)
- Độ dài chuỗi = kích thước mảng – 1

```
char Hoten[30]; // Dài 29 ký tự  
char NgaySinh[9]; // Dài 8 ký tự
```

1.2. Khai báo chuỗi ký tự



Các kiểu khai báo chuỗi

- `char sName[100];`
- `char sName[];`
- `char *sName;`

1.2. Khởi tạo chuỗi ký tự

Khởi tạo như mảng thông thường

- Độ dài cụ thể

```
char s[10] = {'T', 'H', 'C', 'S', ' ', 'A', '\0'};
```

```
char s[10] = "THCS A"; // Tự động thêm '\0'
```

0	1	2	3	4	5	6	7	8	9
T	H	C	S		A	\0			

- Tự xác định độ dài

```
char s[] = {'T', 'H', 'C', 'S', ' ', 'A', '\0'};
```

```
char s[] = 'THCS A'; // Tự động thêm '\0'
```

0	1	2	3	4	5	6
T	H	C	S		A	\0

1.3. Nhập xuất chuỗi



- Hàm nhập chuỗi: **gets**

Ví dụ: `gets(hoten);`

Hàm tự động thêm ký tự NULL (`'\0'`) vào cuối biến chuỗi.

```
void nhapchuoi(char s[100])
{
    printf("Nhap chuoi");
    gets(s); // hàm nhập chuỗi
}
```

1.3. Nhập xuất chuỗi

- Hàm xuất chuỗi: **puts**

Ví dụ: puts(hoten);

```
void Xuatchuoi(char s[100])
{
    printf("Xuatchuoi");
    puts(s); // hàm xuất chuỗi
}
```


1.4. Một số hàm thông dụng trong thư viện



Một số hàm thuộc thư viện <string.h>

- strlen: hàm tính độ dài chuỗi ký tự
- strcpy: hàm sao chép chuỗi ký tự
- strdup: hàm tạo bản sao
- strlwr/strupr: hàm chuyển chuỗi thành chuỗi viết thường / hoa
- strrev : hàm đảo ngược
- strcmp : hàm so sánh 2 chuỗi có phân biệt hoa thường
- stricmp : hàm so sánh 2 chuỗi không phân biệt hoa thường
- strcat : hàm nối 2 chuỗi
- strstr : hàm tìm chuỗi trong chuỗi

2. Các thao tác trên chuỗi ký tự



- 2.1. Đếm các ký tự khoảng trắng trong chuỗi ký tự
- 2.2. Đếm các ký tự hoa / thường trong chuỗi ký tự
- 2.3. Đổi các từ ở đầu câu sang chữ hoa và những từ không phải đầu câu sang chữ thường.
- 2.4. Chuyển các ký tự viết hoa thành viết thường
- 2.5. Chuyển các ký tự viết thường thành viết hoa
- 2.6. Liệt kê các từ trong chuỗi
- 2.7. Xóa các khoảng trắng đầu chuỗi / cuối chuỗi

2.1. Đếm các ký tự khoảng trắng



```
void DemKT(char chuoi[100])
{
    int i;
    int dem=0;
    for (i=0; i<strlen(chuoi); i++)
        if (chuoi[i]==' ')
            dem++;
    return dem;
}
```

2.2. Đếm các ký tự hoa / thường



```
void DemKTThuong(char chuoi[])
{
    int i, dt=0, dh=0;
    for(i=0; i<strlen(chuoi); i++)
        if((chuoi[i]>='a')&&(chuoi[i]<='z'))
            dt++;
        else if ((chuoi[i]>='A')&&(chuoi[i]<='Z'))
            dh++;
    printf("So ky tu thuong: %d", dt);
    printf("So ky tu hoa: %d", dh);
}
```

2.3. Đổi hoa – thường



```
void DoiHoaThuong(char chuoi[100])
{
    chuoi[0]=toupper(chuoi[0]);
    for(int i=1; i< strlen(chuoi); i++)
        chuoi[i]=tolower(chuoi[i]);
    printf("Xuat chuoi");
    puts(chuoi);
}
```

2.4. Chuyển các ký tự viết hoa thành viết thg

```
void ChuyenHoaSangThuong(char chuoi[100])
{
    char kq[100];
    strcpy(kq, chuoi);
    for(int i=0; kq[i]!='\0'; i++)
        if ((kq[i]>='A') && (kq[i]<='Z'))
            kq[i]=tolower(kq[i]);
    printf("Xuat chuoi");
    puts(kq);
}
```

2.5. Chuyển các ký tự viết thg thành viết hoa

```
void ChuyenThuongSangHoa(char chuoi[100])
{
    char kq[100];
    strcpy(kq, chuoi);
    for(int i=0; kq[i]!='\0'; i++)
        if ((kq[i]>='a') && (kq[i]<='z'))
            kq[i]=toupper(kq[i]);
    printf("Xuat chuoi");
    puts(kq);
}
```

2.6. Liệt kê các từ trong chuỗi



```
void LietKe (char chuoi[100])
{
    int d=0;
    for(i=0; i<strlen(chuoi); i++)
        if(chuoi[i]==' ')
        {
            for(j=d; j<i; j++)

                printf("%c",chuoi[j]);
            d=i+1;
            printf("\n");
        }
}
```


2.7. Xóa các khoảng trắng



```
void xoadau (char chuoi[100])
{
    int i=0
    while (chuoi[0]!='\0')
    {
        for(int i = 0; i < strlen(chuoi); ++i)
            str[i] = str[i + 1];
    }
}

void xoacuo (char chuoi[100])
{
    while (chuoi[strlen(chuoi)]!='\0')
        chuoi[strlen(chuoi)]='\0';
}
```



- Nhập / xuất chuỗi
- Xuất các ký tự in hoa trong chuỗi
- Đảo ngược các ký tự trong chuỗi.
- Đổi chữ xen kẽ 1 chữ hoa và 1 chữ thường.
- Đếm một ký tự xuất hiện bao nhiêu lần trong chuỗi.
- Tìm kiếm xem ký tự nào xuất hiện nhiều nhất trong chuỗi.
- Kiểm tra xem chuỗi có đối xứng hay không?
- Nhập vào một từ và xóa từ đó trong chuỗi đã cho.