

Bài tập thực hành về Chuỗi ký tự

Site: [E-Learning Nha Trang University](#)
Course: Nhập môn lập trình - SOT315_66.KHMT - GV.
Lê Thị Bích Hằng
Book: Bài tập thực hành về Chuỗi ký tự

Printed by: Tiến Phạm Minh
Date: Tuesday, 27 May 2025, 10:34 PM

Table of contents

1. Một số chương trình tham khảo

- 1.1. Viết chương trình nhập vào một chuỗi và in ra chuỗi vừa nhập.
- 1.2. Tính độ dài của chuỗi
- 1.3. Xuất từng ký tự của chuỗi
- 1.4. Đếm ký tự nguyên âm trong chuỗi
- 1.5. Kiểm tra chuỗi có đối xứng
- 1.6. Kiểm tra ký tự có nằm trong chuỗi
- 1.7. So sánh hai chuỗi
- 1.8. Chuyển chuỗi thành chữ cái hoa/thường
- 1.9. Nhập và hiển thị danh sách sinh viên
- 1.10. Đếm số từ trong chuỗi
- 1.11. Chuẩn hóa chuỗi
- 1.12. Chuyển một số sang chuỗi

2. Bài tập thực hành

1. Một số chương trình tham khảo

1. Đếm ký tự nguyên âm trong chuỗi.
2. Kiểm tra chuỗi có đối xứng
3. Kiểm tra một ký tự có xuất hiện trong chuỗi hay không, nếu có thì đếm số lần xuất hiện của ký tự.
4. So sánh hai chuỗi.
5. Chuyển đổi chuỗi.
6. Nhập và hiển thị danh sách chuỗi (mảng chuỗi hai chiều).
7. Đếm số từ trong chuỗi
8. Chuẩn hóa chuỗi.

1.1. Viết chương trình nhập vào một chuỗi và in ra chuỗi vừa nhập.

```
#include<stdio.h>
#include<string.h>
int main(){
    char s[100];
    printf("Nhập vào 1 chuỗi: "); //puts("Nhập vào 1 chuỗi: ");
    gets(s);
    printf("Chuỗi vừa nhập vào: ");
    puts(s);
    return 0;
}
```

1.2. Tính độ dài của chuỗi

Viết chương trình nhập vào một chuỗi, sau đó cho biết độ dài của chuỗi nhập vào (không dùng hàm có sẵn `strlen`).

```
#include<stdio.h>
#include<string.h>
int main(){
    char s[100];
    printf("Nhap vao 1 chuoi: ");
    gets(s);
    printf("Chuoi vua nhap vao: ");
    puts(s);
    int i=0;
    while(!s[i]=='\0'){    //while(s[i]!='\0'){
        i++;
    }
    printf("Do dai chuoi = %d",i);
    return 0;
}
```

1.3. Xuất từng ký tự của chuỗi

Viết chương trình nhập vào một chuỗi, sau đó xuất ra từng ký tự trong chuỗi, mỗi ký tự cách nhau 1 khoảng trắng.

```
#include<stdio.h>
#include<string.h>
int main(){
    char s[100];
    printf("Nhap vao 1 chuoi: ");
    gets(s);
    printf("Chuoi vua nhap vao: ");
    int i=0;
    while(!s[i]=='\0'){
        printf("%c ",s[i]);
        i++;
    }
    // for(i=0;i<strlen(s);i++){
    //     printf("%c ",s[i]);
    // }
    return 0;
}
```

*** SV có thể thay đổi đoạn chương trình trên theo yêu cầu như sau: Viết chương trình nhập vào một chuỗi, sau đó xuất ra từng ký tự trong chuỗi, mỗi ký tự nằm trên 1 dòng.

1.4. Đếm ký tự nguyên âm trong chuỗi

Chương trình nhập vào một khối văn bản bất kỳ cho đến khi gặp ký tự '\$' để kết thúc. Đếm số ký tự nguyên âm có trong khối văn bản đã nhập.

- Chương trình:

```
/*Chương trình đếm số ký tự nguyên âm trong chuỗi */  
  
#include<stdio.h>  
  
#include<string.h>  
  
#include<conio.h>  
  
int main()  
{ char txt[255];  
  
int count,i;  
  
printf("Nhập vào khối văn bản. Nhập $ để kết thúc :\n");  
  
i = 0;  
  
while((txt[i++] = getchar( )) != '$');  
  
count = 0;  
  
for(i = 0; i < strlen(txt); i++)  
{ switch(txt[i])  
  
{ case 'a': case 'i': case 'e': case 'o': case 'u':  
  
case 'A': case 'I': case 'E': case 'O': case 'U':  
  
count++;  
  
// break;  
  
}  
  
}  
  
printf("\nSố ký tự nguyên âm = %d", count);  
  
getch();  
  
}
```

1.5. Kiểm tra chuỗi có đối xứng

Nhập chuỗi s từ bàn phím. Kiểm tra tính đối xứng của chuỗi vừa nhập.

- Ý tưởng:
 - Giả sử chuỗi đối xứng,
 - Lần lượt kiểm tra từng cặp ký tự: Nếu có 1 cặp ký tự (i, n - i - 1) khác nhau => chuỗi không đối xứng (n là độ dài của chuỗi).
- Chương trình:

```
#include <stdio.h>
#include <string.h>
#define MAX 100
char KT_Doixung(char s1[])
{
    int i;
    for (i = 0; i < strlen(s1) / 2; i++)
    {
        if (s1[i] != s1[strlen(s1) - 1 - i])
        {
            return 0;
        }
    }
    return 1;
}
int main()
{
    char s[MAX];
    printf("Nhap chuoi: ");
    gets(s);
    if (KT_Doixung(s) == 0)
    {
        printf("Chuoi khong doi xung\n");
    }
    if (KT_Doixung(s) == 1)
    {
        printf("Chuoi doi xung\n");
    }
}
```


1.6. Kiểm tra ký tự có nằm trong chuỗi

Chương trình nhập vào một chuỗi và một ký tự, kiểm tra ký tự có trong chuỗi hay không, nếu có thì cho biết số lần xuất hiện của ký tự đó trong chuỗi.

- Chương trình:

```
#include <stdio.h>
#include <string.h>
#define MAX 100
/*dem ki tu xuat hien trong chuoi*/
int KT_Kytu(char s[], char c)
{
    int i, dem = 0;
    for (i = 0; i < strlen(s); i++)
    {
        if (s[i] == c)
        {
            dem++;
        }
    }
    return dem;
}
int main()
{
    char s[MAX], ch;
    int kq ;
    printf("Nhap chuoi: ");
    gets(s);
    printf("Chuoi vua nhap la : %s", s);
    printf("\nNhap ki tu can dem: ");
    scanf("%c", &ch);
    kq = KT_Kytu(s,ch);
    if (kq)
        printf("\nKy tu %c xuat hien %d lan trong chuoi", ch,kq);
    else
        printf("\nKhong co ky tu %c trong chuoi",ch);
}
```

1.7. So sánh hai chuỗi

Chương trình đoán tên bài hát yêu thích

- Chương trình:

```
#include <stdio.h>

#include <string.h>

int main ()
{
    char key[] = "Heal the world";
    char buffer[80];
    do{
        printf ("Hay doan ten bai hat toi thich? ");
        gets(buffer);
    }while (strcmp (key,buffer) != 0);    //Khong phan biet chu cai hoa - thuong
    puts("Ban doan dung roi!");
    return 0;
}
```

1.8. Chuyển chuỗi thành chữ cái hoa/thường

Chuyển chuỗi nhập từ bàn phím thành chữ viết hoa, viết thường và viết hoa mỗi chữ cái đầu mỗi từ.

- Chương trình:

```
#include <stdio.h>
#include <string.h>
#define MAX 100
void to_lower(char s[]);
void to_upper(char s[]);
void to_proper(char str[]);
////////////////////////////////////
int main()
{
    char s[MAX];
    int i;
    printf("\nNhap chuoi : ");
    gets(s);
    to_lower(s);
    printf("\nChuoi dang chu thuong = %s", s);
    to_upper(s);
    printf("\nChuoi dang chu hoa = %s", s);
    to_proper(s);
    printf("\nChuoi dang viet hoa dau moi tu = %s", s);
    return 0;
}
////////////////////////////////////
void to_lower(char s[])
{
    for (int i = 0; s[i] != '\0'; i++)
    {
        if (s[i] >= 'A' && s[i] <= 'Z')
        {
            s[i] = s[i] + 32;
        }
    }
}

void to_upper(char s[])
{
    for (int i = 0; s[i] != '\0'; i++)
    {
        if (s[i] >= 'a' && s[i] <= 'z')
        {
            s[i] = s[i] - 32;
        }
    }
}

void to_proper(char str[])
{
    for (int i = 0; str[i] != '\0'; i++)
    {
        if (str[i] >= 'A' && str[i] <= 'Z')
            str[i] += 32;
        if (str[i - 1] == ' ' || i == 0)
        {
            if (str[i] >= 'a' && str[i] <= 'z')
                str[i] = str[i] - 32;
        }
    }
}
```


1.9. Nhập và hiển thị danh sách sinh viên

Chương trình nhập vào một danh sách sinh viên, hiển thị danh sách sinh viên ra màn hình.

- Hình dưới đây minh họa mảng chuỗi hai chiều:

Memory location(base address)	Array elements									
25860	t	r	e	e	\0					
25870	b	o	w	l	\0					
25880	h	a	t	\0						
25890	m	i	c	e	\0					
25900	t	o	o	n	\0					

length of each String is [10]

[5] names stored in 5 different memory locations

- Chương trình:

```
#include <stdio.h>
#include <string.h>
#define MAX 100
voidNhap_DS(char arr[][50], int &n);
voidXuat_DS(char arr[][50], int n);
```

```
////////////////////////////////////
int main(){
    char a[MAX][50]; // mảng chứa tối đa 100 sinh viên, mỗi tên sinh viên dài tối đa 50 ký tự
    int n;
    do{
        printf("\nNhập số lượng SV: ");
        scanf("%d", &n);
    }while(n < 1);
    //Nhập danh sách
    Nhap_DS(a,n);
    //Xuất danh sách
    Xuat_DS(a,n);
}
```

```
////////////////////////////////////
voidNhap_DS(char arr[][50], int &n)
{
    for(int i = 0; i < n; i++){
        printf("Tên SV thứ %d: ", i+1);
        fflush(stdin);
        gets(arr[i]);
    }
}
voidXuat_DS(char arr[][50], int n)
{
    for(int i = 0; i < n; i++){
        printf("\nTên SV thứ %d: %s", i+1, arr[i]);
    }
}
```

1.10. Đếm số từ trong chuỗi

Chương trình đếm số từ có trong một chuỗi được nhập từ bàn phím.

- Ý tưởng:
 - Nếu ký tự hiện tại là dấu cách và ký tự tiếp theo không phải dấu cách thì ta coi đó là bắt đầu của 1 từ.
- Lưu ý:
 - Có thể từ đầu tiên không bắt đầu bằng dấu cách.
- Chương trình:

```
#include <stdio.h>
#include <string.h>
#define MAX 100
int demTu(char str[])
{
    int count;
    count = (str[0] != ' ');
    for (int i = 0; i < strlen(str) - 1; i++)
    {
        if (str[i] == ' ' && str[i + 1] != ' ')
        {
            count++;
        }
    }
    return count;
}
int main()
{
    char s[MAX];
    printf("Nhap chuoi: ");
    gets(s);
    printf("\nChuoi co %d tu", demTu(s));
}
```

1.11. Chuẩn hóa chuỗi

Chương trình chuẩn hóa một chuỗi bất kỳ

- Ý tưởng:
 - Trình tự chuẩn hóa một chuỗi như sau:
 1. Bỏ các ký tự khoảng trắng thừa ở đầu chuỗi;
 2. Bỏ các ký tự khoảng trắng thừa ở cuối chuỗi;
 3. Viết hoa chữ cái đầu tiên của chuỗi;
 4. Đưa các ký tự còn lại về dạng chữ thường;
 5. Bỏ qua các khoảng trắng thừa ở giữa;
 6. Viết hoa ký tự đầu mỗi từ.
- Chương trình:

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
#define MAX 100

void chuanHoaChuoi(char str[], char s[]) //str la chuoi ban dau, s la chuoi ket qua
{
    int first = 0, last = strlen(str);
    int count = 0;
    // Xoa ky tu trang dau chuoi
    while (first < last && str[first] == ' ')
        first++;

    // Xoa ky tu trang cuoi chuoi
    while (last > first && str[last] == ' ')
        last--;

    // Viet hoa chu cai dau tien
    if (str[first] >= 'a' && str[first] <= 'z')
    {
        str[first] -= 32;
        s[0] = str[first];    //Copy ky tu dau tien cua chuoi str vao chuoi s
    }

    for (int i = first + 1; i <= last; i++)
    {
        // Chuyen tat ca chu cai con lai thanh chu thuong
        if (str[i] >= 'A' && str[i] <= 'Z')
            str[i] += 32;

        if (i + 1 <= last)
        {
            //Bo dau trang thua o giua
            if (str[i] == str[i + 1] && str[i] == ' ')
                continue;
        }
        // Copy ky tu str[i] vao chuoi s
        s[++count] = str[i];
    }
    for(int i=0; i<strlen(s);i++)
    {
        // Viet hoa ky tu dau moi tu cua chuoi s
        if (s[i] == ' ' && s[i + 1] >= 'a' && s[i + 1] <= 'z')
            s[i + 1] -= 32;
    }
}

int main()
{
    char s[MAX], s1[MAX];
    printf("Nhap chuoi: ");
    gets(s);
    printf("\nChuoi sau khi chuan hoa: ");
    chuanHoaChuoi(s,s1);
    puts(s1);
    getch();
}
```


1.12. Chuyển một số sang chuỗi

Chương trình chuyển một số nguyên sang dạng chuỗi

- Sử dụng hàm itoa trong thư viện <stdlib.h>
- Chương trình:

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main ()
```

```
{
```

```
    int i;
```

```
    char buffer[50];
```

```
    printf ("Nhap vao mot so nguyen: ");
```

```
    scanf ("%d",&i);
```

```
    itoa (i,buffer,10);
```

```
    printf ("Dang thap phan: %s\n",buffer);
```

```
    itoa (i,buffer,16);
```

```
    printf ("Dang thap luc phan: %s\n",buffer);
```

```
    itoa (i,buffer,2);
```

```
    printf ("Dang nhi phan: %s\n",buffer);
```

```
    return 0;
```

```
}
```

2. Bài tập thực hành

I. BÀI TẬP CƠ BẢN

Bài 1. Viết chương trình nhập vào một chuỗi và in ra chuỗi vừa nhập.

Bài 2. Viết chương trình nhập vào một chuỗi, sau đó in ra chuỗi và cho biết độ dài của chuỗi nhập vào (không dùng hàm có sẵn `strlen`).

Bài 3. Viết chương trình nhập vào một chuỗi, sau đó xuất ra từng ký tự trong chuỗi, mỗi ký tự cách nhau 1 khoảng trắng.

Bài 4. Viết chương trình nhập vào một chuỗi, sau đó xuất ra chuỗi theo thứ tự đảo ngược (dùng hàm có sẵn hoặc không dùng hàm).

Bài 5. Viết chương trình nhập vào một chuỗi, sau đó cho biết số ký tự chữ cái, số ký số và số ký tự đặc biệt có trong chuỗi.

Bài 6. Viết chương trình nhập vào một chuỗi, sau đó sắp xếp chuỗi theo thứ tự tăng dần. In ra chuỗi sau khi sắp xếp.

II. BÀI TẬP TỔNG HỢP

Bài 1a. Viết chương trình nhập vào 1 chuỗi và 1 ký tự, kiểm tra ký tự có trong chuỗi hay không, nếu có đưa ra số lần xuất hiện của ký tự đó trong chuỗi.

Ví dụ: Chuỗi = "Nhap mon lap trinh" ; ký tự = 'a'

In ra: Có ký tự 'a' trong chuỗi. Số lần xuất hiện = 2

Bài 1b. Viết chương trình nhập vào một chuỗi tiếng Anh, sau đó cho biết số lần xuất hiện từ "the" có trong chuỗi (Từ "the" không phân biệt chữ hoa-thường).

Ví dụ: Chuỗi nhập vào: The earth, the sun, the moon and THE sky

In ra: Từ "the" xuất hiện 4 lần trong chuỗi

Bài 2. Viết chương trình nhập vào một chuỗi:

a. In ra các ký tự lên màn hình, mỗi ký tự nằm trên một dòng.

b. Đếm số ký tự trong chuỗi.

c. Đếm số ký tự không phải nguyên âm trong chuỗi.

Bài 3. Viết chương trình nhập vào một chuỗi. Hiển thị chuỗi theo thứ tự đảo ngược.

Bài 4. Viết chương trình thay thế các ký tự không phải nguyên âm trong chuỗi đã cho thành khoảng trắng.

Bài 5. Viết chương trình nhập một chuỗi từ bàn phím. Kiểm tra tính đối xứng của chuỗi vừa nhập.

Ví dụ: Chuỗi `abcdcba` là chuỗi đối xứng; chuỗi `abcd` không phải chuỗi đối xứng

Gợi ý: Sử dụng vòng lặp để duyệt chuỗi. Nếu có 1 cặp ký tự ($i, n - i - 1$) khác nhau \Rightarrow chuỗi không đối xứng (n là độ dài của chuỗi).

Bài 6. Viết chương trình nhập vào một chuỗi, sau đó loại bỏ những khoảng trắng thừa trong chuỗi.

Bài 7. Viết chương trình nhập vào một chuỗi (hai từ chỉ cách một khoảng trắng):

- a. Đếm số từ có trong chuỗi.
- b. In ra từ đầu tiên.
- c. In ra từ cuối cùng.
- d. In ra các từ trong chuỗi, mỗi từ nằm trên một dòng.

Bài 8. Viết chương trình nhập một chuỗi từ bàn phím. Sau đó:

- a. Chuyển các ký tự chữ cái trong chuỗi thành chữ in hoa (không dùng hàm `strupr`)
- b. Chuyển các ký tự chữ cái trong chuỗi thành chữ in thường (không dùng hàm `strlwr`)
- c. Chuyển mỗi chữ cái đầu mỗi từ thành chữ in hoa, các chữ cái còn lại trong mỗi từ thành chữ in thường.

Bài 9. Viết chương trình nhập vào 2 chuỗi `s1` và `s2`. Tìm kiếm chuỗi `s1` có nằm trong chuỗi `s2`, nếu có thì cho biết vị trí xuất hiện của chuỗi `s1` trong `s2`.

Bài 10. Viết hàm tra xem trong chuỗi có ký tự số hay không, nếu có thì sao chép các ký số vào một mảng số riêng.

Bài 11. Viết chương trình nhập một chuỗi bất kỳ, yêu cầu nhập 1 ký tự muốn xóa. Thực hiện xóa tất cả những ký tự đó trong chuỗi.

Bài 12. Cho chuỗi `str`, nhập vào vị trí `vt` và số ký tự cần xóa `n`, hãy xóa `n` ký tự tính từ vị trí `vt` trong chuỗi `str`.

Bài 13. Viết chương trình tìm kiếm xem ký tự nào xuất hiện nhiều nhất trong chuỗi.

Bài 14. Viết chương trình nhập một chuỗi từ bàn phím (Các từ cách nhau một khoảng trắng). Tìm từ có chiều dài dài nhất và in ra.

Bài 15. Viết chương trình nhập một chuỗi từ bàn phím (Các từ cách nhau một khoảng trắng). Tìm từ có chiều dài ngắn nhất và in ra.