**字符串链接：**

char \*MyStrcat(char \*str1, char \*str2)

{

char \*pStr = str1;

while (\*str1 != '\0')

{

str1++;

}

for(; \*str2 != '\0'; str1++,str2++)

{

\*str1 = \*str2;

}

\*str1 = '\0';

return pStr;

}

**字符串逆序存放：**

#include <stdio.h>

#include <string.h>

#define N 80

char \*InvertStr(char \*str, int n);

int main()

{

int n;

char \*tag = NULL;

static char str[N];

printf("Input a string:");

gets(str);

n = strlen(str);

tag = InvertStr(str, n);

printf("Inversed results:%s\n", tag);

return 0;

}

char \*InvertStr(char \*str, int n)

{

int i;

static char ivstr[N];

for (i = 0, str = str+n-1; n >= 0; i++, str--, n--)

{

ivstr[i] = \*str;

}

return ivstr;

}

**统计字符串长度：**

#include <stdio.h>

#define N 80

unsigned int MyStrlen( char \*pstr);

int main()

{

int n;

static char str[N];

printf("Please enter a string:\n");

gets(str);

n = MyStrlen(str);

printf("The length of the string is:%u\n", n);

return 0;

}

unsigned int MyStrlen( char \*pstr)

{

int i;

for (i = 0; \*pstr != '\0'; i++, pstr++)

{

;

}

return i;

}

**删除指定字符：**

#include <stdio.h>

#include <stdlib.h>

#define N 80

void RemoveRepeatStr(char \*str, char ch);

char str[N];

int main()

{

char ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

scanf("%s", &ch);

RemoveRepeatStr(str, ch);

printf("Results:%s\n", str);

return 0;

}

void RemoveRepeatStr(char \*str, char ch)

{

char \*p = NULL;

for (; \*str != '\0'; str++)

{

if (\*str == ch)

{

p = str;

for (; \*p != '\0'; p++)

{

\*p = \*(p + 1);

}

str--;

}

}

}