

```

#include <cstdio>
#include <iostream>
#include <cstring>
#include <string>

using namespace std;
void char_array(){
    // 字符数组
    /*
    1. 字符串一定是字符数组 但是字符数组不一定是字符串
    2. 末尾是\0
    */
    char s2[5] = {'H', 'e', 'l', 'l', 'o'};
    char s1[6] = {'H', 'e', 'l', 'l', 'o', '\0'}; // 字符串 '\n' '\t' 代表了字符串的结尾
    char s[] = "Hello"; // 字符串 它会自动在后面加 \0

    char s4[6] = {'H', 'e', 'l', 'l', 'o'}; //

    char test[26] = {'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K',
                    'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V',
                    'W', 'X', 'Y', 'Z'}; // 字符数组
    char test2[] = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"; // 字符串
    for (int i = 0; i < 27; i++){
        printf("%d ", test[i]);
    } // 可以发现最后一个数字是随机的
    cout << endl;
    for (int i = 0; i < 27; i++){
        printf("%d ", test2[i]);
    } // 可以发现最后一个数字是0
}

void print_char_array(){
    /*
    putchar 打印一个字符 和 printf("%c", s[i]);功能一样
    */
    char s[] = "Hello";
    for(int i = 0; i < 5; i++){
        putchar(s[i]);
    }
    // 这三种方法都不会去打印最后的\0
    printf("%s", s); // s->string(字符串) c的方法 不会换行
    cout << s;      // c++的方法 不会换行
    puts(s);        // c的方法 会在最后换行 (\n)
}

```

/\*cspj 第一轮 70

第二轮 四道编程题 150/400 科技特长生的要求 4小时 体验一下

```

noip 中级的水平
noi 全国奥林匹克竞赛的全国选拔国家训练营
*/

void read_char_array(){
    int x[] = {1,2,3};

    char s[10];
    /*
        注意这里s已经是地址了所以不用加&
        scanf 遇到 空格 tab 换行符 输入会结束
        输入 abc def
        输出 abc
    */
    scanf("%s", s);
    cout << s << endl;

    cin >> s;
    cout << s << endl; // scanf cin 如果遇到空格的话 空格 tab 回车的后面的字符无法被写入

    /*
        常用的字符串输入方式
        遇到空格tab换行均不会结束输入 只有遇到\0才会结束输入
    */
    fgets(s, 10, stdin);
    cout << s << endl;

    /*
        gets 是fgets的前身 现在已经不用了
    */
    // gets(s);
    // cout << s << endl;
}

void read_file(){
    FILE *file;
    char lines[20]; // Buffer to store each line
    file = fopen("example.txt", "r");
    if (file == NULL) {
        perror("Error opening file");
        return;
    }
    /*
        了解即可
        知道fgets不仅可以从键盘输入还可以通过文件读取
    */
    while (fgets(lines, sizeof(lines), file) != NULL) {
        printf("%s", lines); // Print each line
    }
}

```

```
}  
    // Close the file  
    fclose(file);  
}  
  
int main(){  
  
    char_array();  
    print_char_array();  
    read_char_array();  
    read_file();  
    // str_utils();  
    // str();  
}
```