

Welcome

数据科学与大数据技术专业 程序设计基础(C语言)

上海体育学院经济管理学院

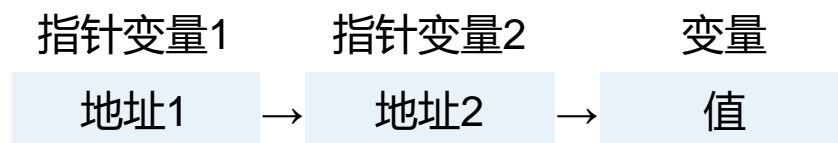
Wu Ying

指向指针的指针变量

如果在一个指针变量中存放一个目标变量的地址，这就是“单级间址”；



指向指针数据的指针用的是“二级间址”方法；



取内容——“解引用”

利用二级指针动态建立二维数组

```

1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <string.h>
5  #include <time.h>
6  int** MallocArray(int row, int col);
7  void DeleteArray(int** p, int row);
8  void ShowArray(int** p, int row, int col);
9  int main(void)
10 {
11     int nr, nc; //行数, 列数
12     int** p;
13     nr = 5;
14     nc = 4;
15
16     p = MallocArray(nr, nc);
17     ShowArray(p, nr, nc);
18     DeleteArray(p, nc);
19
20     return 0;
21 }

```

```

22 int** MallocArray(int row, int col)
23 {
24     int i, j;
25     int** pa;
26     srand((unsigned)time(NULL));
27     pa = (int**)malloc(row * sizeof(int*));
28     for (i = 0; i < row; i++)
29     {
30         *(pa + i) = (int*)malloc(col * sizeof(int));
31     }
32
33     for (i = 0; i < row; i++)
34     {
35         for (j = 0; j < col; j++)
36         {
37             (*(pa + i) + j) = rand() % 21;
38         }
39     }
40
41     return pa;
42 }

```

```

43 void DeleteArray(int** p, int row)
44 {
45     int i;
46     for (i = 0; i < row; i++)
47         free(p[i]);
48     free(p);
49 }
50 void ShowArray(int** p, int row, int col)
51 {
52     int i, j;
53     for (i = 0; i < row; i++)
54     {
55         for (j = 0; j < col; j++)
56         {
57             printf("%d\t", *(p + i) + j);
58         }
59         printf("\n");
60     }
61 }
62 }

```

先申请一维的指针数组，然后该数组中的每个指针再申请数组，相当于二维数组

但是这种方法会导致每行可能不相邻，从而访问效率比较低。

Microsoft Visual Studio 调试控制台

```

4      17      12      13
19      5       2      14
9       0      10      6
11      18      16      19
5       12      8       20

```

返回指针值的函数

类型名 *函数名(参数表列)

一个函数可以返回一个整型值、字符值、实型值等，也可以返回指针型的数据

```
int *fun(int x,int y);
```

定义和使用结构体变量

自己建立结构体类型

(1) 结构体类型并非只有一种，而是可以设计出许多种结构体类型，各自包含不同的成员。

(2) 成员可以属于另一个结构体类型。

| num | name | sex | age | birthday | | | addr |
|-----|------|-----|-----|----------|-----|------|------|
| | | | | month | day | year | |

```
struct Date //声明一个结构体类型 struct Date
{
    int month; //月
    int day; //日
    int year; //年
};
```

```
struct Student //声明一个结构体类型 struct Student
{
    int num;
    char name[20];
    char sex;
    int age;
    struct Date birthday; //成员birthday属于struct Date类型
    char addr[30];
};
```

定义结构体类型变量

1. 先声明结构体类型，再定义该类型的变量

```
struct Student
{
    int num;           //学号为整型
    char name[20];     //姓名为字符串
    char sex;          //性别为字符型
    int age;           //年龄为整型
    float score;       //成绩为实型
    char addr[30];     //地址为字符串
};                    //注意最后有一个分号
```

```
struct Student student1, student2;
```

结构体类型名 结构体变量名

| | | | | | | |
|-----------|-----------|--------------|---|--------|----------|--------------|
| student1: | 1000 1 | Zhang Xin | M | 1 9 | 90. 5 | Shangha i |
| student2: | 1000 2 | Wang Li | F | 2 0 | 98 | Beijing |

2. 在声明类型的同时定义变量

```
struct Student
{
    int num;
    char name[20];
    char sex;
    int age;
    float score;
    char addr[30];
}student1, student2;
```

struct 结构体名
{ 成员表列
}变量名表列;

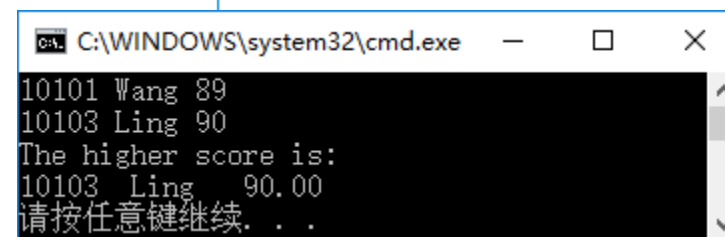
3. 不指定类型名而直接定义结构体类型变量

struct
{ 成员表列
}变量名表列;

结构体变量的初始化和引用

【例9.2】输入两个学生的学号、姓名和成绩，输出成绩较高的学生的学号、姓名和成绩。

```
#include <stdio.h>
int main()
{
    struct Student          //声明结构体类型struct Student
    {
        int num;
        char name[20];
        float score;
    }student1,student2;    //定义两个结构体变量student1,student2
    scanf("%d%s%f",&student1.num,student1.name,&student1.score); //输入学生1的数据
    scanf("%d%s%f",&student2.num,student2.name,&student2.score); //输入学生1的数据
    printf("The higher score is:\n");
    if(student1.score>student2.score)
        printf("%d %s %6.2f\n",student1.num,student1.name,student1.score);
    else if(student1.score<student2.score)
        printf("%d %s %6.2f\n",student2.num,student2.name,student2.score);
    else
    {
        printf("%d %s %6.2f\n",student1.num,student1.name,student1.score);
        printf("%d %s %6.2f\n",student2.num,student2.name,student2.score);
    }
    return 0;
}
```



```
C:\WINDOWS\system32\cmd.exe
10101 Wang 89
10103 Ling 90
The higher score is:
10103 Ling 90.00
请按任意键继续. . .
```


c读写文件 —— 体验何为读写数据库

| H | T | R | La | w |
|-------|------|----|--------|----------|
| 7.13 | 7.13 | 35 | 687.61 | 0.00351 |
| 6.68 | 7.32 | 35 | 426.8 | 0.00182 |
| 5.76 | 4.65 | 35 | 666.4 | 0.00792 |
| 2.65 | 4.62 | 35 | 554.78 | 0.000854 |
| 10.82 | 4.98 | 35 | 423.24 | 0.000364 |
| 10.13 | 4.68 | 35 | 504.61 | 0.000115 |

fseek(文件指针, 位移量, 起始位置);

| 整数 | 符号常量 | 对应的起始位置 |
|----|----------|-----------|
| 0 | SEEK_SET | 文件开头 |
| 1 | SEEK_CUR | 文件指针的当前位置 |
| 2 | SEEK_END | 文件末尾 |

```
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3  void main()
4  {
5      int i, j;
6      float da[6][5] = { 0 };
7
8      char chy[4] = { 'x', 'a', 'h', 'w' };
9      int data[4] = { 1, 3, 6, 9 };
10
11
12      FILE* fp = NULL;
13
14      fp = fopen("read.csv", "r");
15      //读文件
16      for (i = 0; i < 6; i++)
17          for (j = 0; j < 5; j++)
18              {
19                  fscanf(fp, "%f", &da[i][j]);
20                  fseek(fp, 1L, SEEK_CUR); /*fp指针从当前位置向后移动*/
21              }
22      //显示
23      for (i = 0; i < 6; i++)
24      {
25          for (j = 0; j < 5; j++)
26              printf("%f\t", da[i][j]);
27          printf("\n");
28      }
29
30      //写文件
31      fp = fopen("read2.csv", "w");
32      fseek(fp, 0L, SEEK_END);
33      for (i = 0; i < 4; i++)
34          fprintf(fp, "%c\t%d\n", chy[i], data[i]);
35
36      fclose(fp); /*释放空间*/
37
38  }
```

课后题6：写一个函数，求一个字符串的长度，在main函数中输入字符串，并输出其长度。

课后题6: 写一个函数, 求一个字符串的长度, 在main函数中输入字符串, 并输出其长度。

```
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3
4  int length(char* p);
5  int main()
6  {
7      char str[50];
8      int len;
9      printf("input string: ");
10     gets_s(str, 49);
11     len = length(str);
12     printf("The length of string is %d. \n", len);
13
14     return 0;
15 }
16 int length(char* p)
17 {
18     int n = 0;
19     while ((*p++) != '\0')
20         n++;
21     return n;
22 }
```

Microsoft Visual Studio 调试控制台

```
input string: Qingyuanhuan Road 650.
The length of string is 22.
```

课后题7:

有一字符串，包含n个字符。写一函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串

课后题7:

有一字符串，包含n个字符。写一函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串

```
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3
4  int length(char* p);
5  void copyfromloc(char* p1, char* p2, int m);
6  int main()
7  {
8      char str[50];
9      char dt[50];
10     int loc;
11     printf("input string: ");
12     gets_s(str, 49);
13     printf("input the location to copy:");
14     scanf("%d", &loc);
15     if (length(str) < loc)
16         printf("input error!起始位置大于字符串长度\n");
17     else
18     {
19         copyfromloc(str, dt, loc);
20         printf("结果: %s\n", dt);
21     }
22     return 0;
23 }
24 int length(char* p)
25 {
26     int n = 0;
27     while ((*p++) != '\0')
28         n++;
29     return n;
30 }
```

```
31 void copyfromloc(char* p1, char* p2, int m)
32 {
33     int i = 0;
34     while (i < m - 1)
35     {
36         i++;
37         p1++;
38     }
39     while (*p1 != '\0')
40     {
41         *p2 = *p1;
42         p1++;
43         p2++;
44     }
45     *p2 = '\0';
46 }
```

Microsoft Visual Studio 调试控制台

```
input string: Qingnyuanhuan Road 650.
input the location to copy:9
结果: nhuan Road 650.
```

使用easyx中的putimage函数实现无背景贴图

<https://blog.csdn.net/xiongzebao/article/details/44713265>