**1、c语言指针篇——数组指针和指向二维数组的指针变量（\*p）[N]**

<https://blog.csdn.net/qq_36744540/article/details/79832728>

**2、C语言----指针形参(指向指针的指针形参)**

<https://blog.csdn.net/a3748622/article/details/79286974>

# 3、二维字符数组的定义，初始化及引用，C语言二位字符数组详解

<http://c.biancheng.net/view/273.html>

# 4、二维数组作为形参--传值的3三种方式

<https://blog.csdn.net/YunLaowang/article/details/82054680>

# 5、”指向指针的指针“作用与应用

<https://blog.csdn.net/deeplan_1994/article/details/82458098>

int a[4][2] = { { 2,4 },{ 6,8 },{ 1,3 },{ 5,7 } };

char \*p[5] = { "acvsadf", "cwerqwenn", "tttttt" };//字符串指针数组

int c[4] = { 0,1,2,3 };

int \*ptr = c;

/\*

这个正确，说明a[0]是一个地址，而且只指向一个int类型；

也说明它是一维数组（这个数组有两个元素，即2和4）的首元素（这里是2）的地址；

+1会加上4个字节，即一个int类型的大小；

\*/

int \*p1 = a[0];

/\*

这个正确，说明&a[0]是一个地址，而且指向一个包含两个int元素的数组；

也说明它是数组（这个数组有4个元素，分别{ 2,4 },{ 6,8 },{ 1,3 },{ 5,7 }）首元素（这里是{2，4}）的地址；

+1会加上8个字节，即两个int类型的大小；

\*/

int(\*p3)[2] = &a[0];//[]表示的是该指针指向数组，而数组里面有[?]个元素

int(\*p4)[2] = a;//这个说明&a[0]和a性质是一样的。

//int (\*p5)[8] = &a;//error

int\* p8 = (int\*)a;//这样就达到了对整个数组的访问，这种访问方式和访问一维数组无异。

例子：

1、#include <stdio.h>

#include <stdlib.h>

char\* longestCommonPrefix(char\*\* strs, int strsSize) {

if (strs == NULL || \*strs == NULL)

return "";

int len = strlen(strs[0]);

int max = 0,i;

char\* ps = strs[0], \*ret;

while(max < len) {

for (i = 1; i < strsSize; i++) {

if (strs[i][max] != strs[0][max]) {

len = max;

break;

}

}

if (max < len) max++;

}

ret = (char\*)malloc(sizeof(char)\*(max+1));

for (i = 0; i < max; i++) ret[i] = strs[0][i];

ret[max] = '\0';

return ret;

}

int main()

{

/\*

//行指针

char a[3][7] = {"flower","flow","flowe"};

char \*t;

char(\*p)[6];

p=a;//p=&a[0];

\*/

//列指针

char a[3][7]={"flower","flow","flowe"};

char \*b[3]={a[0],a[1],a[2]};

char \*p;

p=&a[0][0];//p=a[0]

char \*t;

t = longestCommonPrefix(b,3);

printf("ret = %s\n",t);

return 0;

}

2、int main()

{

/\*

//行指针

char a[3][7] = {"flower","flow","flowe"};

char \*t;

char(\*p)[6];

p=a;//p=&a[0];

\*/

//列指针

char a[3][7]={"flower","flow","flowe"};

// char \*b[3]={a[0],a[1],a[2]};

char \*p;

p=&a[0][0];//p=a[0]

char \*t;

t = longestCommonPrefix(&p,3);//可以编译通过，但是运行不下去

printf("ret = %s\n",t);

return 0;

}

3、

#include <stdio.h>

#include <stdlib.h>

char\* longestCommonPrefix(char\*\* strs, int strsSize) {

if (strs == NULL || \*strs == NULL)

return "";

int len = strlen(strs[0]);

int max = 0,i;

char\* ps = strs[0], \*ret;

while(max < len) {

for (i = 1; i < strsSize; i++) {

if (\*(\*(strs+i)+max) != \*(\*(strs+0)+max)) {

len = max;

break;

}

}

if (max < len) max++;

}

ret = (char\*)malloc(sizeof(char)\*(max+1));

for (i = 0; i < max; i++) ret[i] = strs[0][i];

ret[max] = '\0';

return ret;

}

int main()

{

/\*

//行指针

char a[3][7] = {"flower","flow","flowe"};

char(\*p)[6];

p=a;//p=&a[0];

\*/

//列指针

char a[3][7]={"flower","flow","flowe"};

char \*b[3]={a[0],a[1],a[2]};

char \*p;

p=&a[0][0];//p=a[0]

char \*t;

t = longestCommonPrefix(&p,3);//这样也可以，但是去掉char \*b[3]={a[0],a[1],a[2]};后就不行，不知道为什么？

printf("ret = %s\n",t);

return 0;

}

**5、求解二维数组的行列数**

对于二维数组： int a[2][5];

求数组元素的总数：sizeof(a)/sizeof(int)

数组行数：(sizeof(a)/sizeof(int))/(sizeof(a[0])/sizeof(int))

数组列数：(sizeof(a[0])/sizeof(int))