**程设作业第6章**

附加1.

#include <stdio.h>

#include <stdlib.h>

#define M 3

#define N 4

typedef int (\*P2D)[M];

void main\_v2() {

int i, j, a[M\*N], (\*p)[N];

for(i = 0; i < M \* N; i++) a[i] = i + 1;

p = (P2D)a;

for(i = 0; i < M; i++) {

for(j = 0; j < N; j++) {

printf("%4d", p[i][j]);

}

printf("\n");

}

}

void f() {

int i, a[N], \*p;

for(i = 0; i < N; i++) {

a[i] = i + 1;

}

p = a;

for(i = 0; i < N; i++) {

printf("%4d", p[i]);

}

}

void main() {

int i, j, a[M\*N], \*p[M];

for(i = 0; i < M \* N; i++) a[i] = i + 1;

for(i = 0; i < M ; i++) {

p[i] = &a[N \* i];

}

for(i = 0; i < M; i++) {

for(j = 0; j < N; j++) {

printf("%4d", p[i][j]);

}

printf("\n");

}

main\_v2();

}

附加2：

概念：创建变量（不包括函数指针）：

1. 创造一个变量
   * 1. 基本格式，\*\*\* a\*\*\* === Type a; (其中\*\*\* ()\*\*\* 表示为Type, 不仅限于基本类型，例int(\*a)[5] 中 Type 表示 int(\*)[5])
     2. 引申，\*\*\* (\*a)\*\*\* === Type (\*a);
        1. a类型推导，Type (\*a) -> (\*a)为Type类型 -> a的Type类型的指针
2. 创造一组变量
   * 1. \*\*\* a[N]\*\*\* === Type a[N]; 创建N个变量，变量类型为 Type, 但是a实际上是首元素的地址。
        1. Type a[N], 首先创建N个变量，其次将a[N]视为整体A, 即Type A, 数组元素类型与A相同

例子：

1. Int a[N][M]://创建N\*M个元素
   1. N 个 类型与int B[M]中B相同的元素（下同）
      1. M 个 int C
2. Int (\*p)[M]://创建一个元素
   1. 一个指针，指向类型 int A[M]
      1. M 个 int B
3. Int \*p[M]://创建M个元素
   1. M 个 int \*A
      1. 一个指针， 指向 int B
4. Int (\*p[N])[M];//创建N个元素
   1. N 个 int (\*A)[M];
      1. 一个指针，指向int B[M]
         1. M个int C

问题分析：

1. 附加1，为什么Main()中需要赋值M次，main\_v2()赋值1次？
2. 上面例子1, 2中a, p类型是什么关系？