实验六：数组(2 学时)

一、实验方式：上机

二、实验目的：

1、掌握一维数组的定义、赋值、输入和输出的方法。
2、掌握一维数组与 for 循环搭配。
3、掌握一维数组与函数搭配。
4、掌握一维数组与宏常量搭配。
5、熟悉查找数组元素—遍历（线性查找）。

三、实验内容及其步骤：

1、要求用户录入 5 个同学的成绩，然后反向顺序输出这些成绩。输入 88 91 95 90 60 100，反向输出 100 60 90 95 91 88。

```
#include<stdio.h>
int main(void)
{
	int num[5], i;
	printf("输入 5 个同学的成绩：");
	for (i = 0; i < 5; i++)
	{
		scanf("%d", &num[i]);
	}
	printf("逆序输出这 5 个同学的成绩：");
	for (i = 4; i >= 0; i--)
	{
		printf("%d  ", num[i]);
	}
	return 0;
}
```

2、在实际开发中，经常需要查询数组中的元素。例如，学校为每位同学分配了一个唯一的学号，现在有一个数组，保存了考研班所有同学的学号信息，有辅导员想知道他分管的学生是否进入了考研班，只需提供要查学生的学号即可，如果辅导员提供的学号和数组中某个学号一致，就说明该生在考研班，否则就不在。不幸的是，C 语言标准库没有提供与数组查询相关的函数，所以我们只能自己写代码。请编程实现这种功能，为简单起见，定义该考研班学生人数有 10 人，每个人的学号 8 位格式 19010101，19 表示年级，01 表示学院，接下依次是班级 01，个人编号 01。该数组是无序数组，用**循环遍历数组中每个元素**，即，把要查询的值挨个比较一遍。

```
#include<stdio.h>
int main(void)
{
	int id[10] = {
	19030102, 19040201, 19010201,
	19040103, 19010102, 19020102,
	19010101, 19060120, 19031031,
	19020131 };
	int i, num, flag = 1;
	//注意：此处又用到了标志变量，如果不用标志变量，能不能实现同样的效果？试一下
	printf("\n***************待查询 8 位学号*****************\n");
	scanf("%d", &num);
	printf("\n***************查询后的结果是***************\n\n");
	for (i = 0; i < 10; i++)
	{
		if (id[i] == num)
		{
			printf("学号%d 在实验班，在第%d 个数组元素！\n", num, i+1);
			flag = 0;
			break;
		}
	}
	if (flag)
	{
		printf("学号%d 不在实验班！\n", num);
	}
	return 0;
}
//注意：这个题本质跟打印素数很像。也可以用此法来查找最大值最小值
```

3、编写程序从键盘输入 19 级某个班同学的 C 语言成绩（整数），然后输出他们的平均值（保留 2 位小数）。要求 1：综合运用之前的函数等知识；要求 2：引入宏常量，每个班同学不超过 10 人。

```
#include<stdio.h>
#include<stdlib.h>
#define N 10
//最大的数组长度为 10，如果同学数超出 10，需要改 N 为更大的长度。
void EnterScore(int score[], int n);
void ReadScore(int score[], int n);
float Average(int score[], int n);

int main(void)
{
    int n, score[N];                                    //内存中留出长度为 N 的内存（4N 个字节）
    printf("请输入 19 级某班的人数为：");
    scanf("%d", &n);
    if (n > N)                                          //检查 n 是否超出 N 的范围
    {
        printf("Error, Please check array[N]\n!");
        exit(0);
    }
    printf("\n***************请开始录入成绩*****************\n");
    EnterScore(score, n);
    printf("\n***************成绩依次排列是*****************\n");
    ReadScore(score, n);
    printf("\n***************最终的平均成绩*****************\n");
    printf("%.2f\n", Average(score, n));
    return 0;
}

void EnterScore(int score[], int n)                     //录入成绩
{
    int i;
    for (i = 0; i < n; i++)
    {
        printf("请输入第%d 个同学的成绩：", i + 1);
        scanf("%d", &score[i]);
    }
}

void ReadScore(int score[], int n)                      //读取成绩
{
    int i;
```

```
    for (i = 0; i < n; i++)
    {
        printf("%d   ", score[i]);                    //%d 后有俩空格，为了输出好看
    }
    printf("\n");
}

float Average(int score[], int n)                     //计算平均值（小数）
{
    int i, sum = 0;
    if (n <= 0)                                       //除 0 错误检查
    {
        printf("Error\n");
        exit(0);
    }
    for (i = 0; i < n; i++)
    {
        sum += score[i];
    }
    return (float)sum /n;                             //强制类型转换
}
```

//注意：上课强调过定义一维数组时候方括号内只能放常量 int num[N]；而实际上这只是针对 C89 的标准。在 C99 标准中，我们定义一维数组 int num[i]是可以在方括号内放变量的，这部分知识属于 C99 的变长数组。这样做的好处是不必让程序员来指定长度，因为数组长度过长，导致浪费内存，例如我们定义的 N=10；int num[N]；意味着我们划出一块内存给这个数组使用，内存总共 4（字节）×10（数组长度）= 40 字节，如果输入学生的数量为 6，实际上是浪费了 4×4=16 个字节的内存；如果数组长度过短，肯定会导致程序程序出错。在本例中，可以尝试将 n 输入 11，超出了我们一开始定义数组内存的大小，运行下会发现程序出错。