



# 作业8

## 一、选择题（每题5分，共20分）

1. 以下关于数组初始化的语句，正确的是？

- A. `int arr[] = {1,2,3};`
- B. `int arr[3] = {1,2,3,4};`
- C. `int arr[];`
- D. `int arr[2] = {};`

答案：A

解析：B选项越界，C未指定大小，D在C语言中非法（C++允许）。

2. 若有定义 `int a[5] = {1,2};`，则 `a[0]` 的值是？

- A. 0
- B. 1
- C. 随机值
- D. 编译错误

答案：B

解析：`a[0]` 被初始化为1

3. 以下代码的输出是？

```
static int arr[3] = {5};  
printf("%d", arr[1]);
```

- A. 5
- B. 0
- C. 随机值
- D. 编译错误

答案：B

解析：静态数组部分初始化时，未指定的元素为0。

4. 数组下标允许的范围是？

- A. 从1到声明的大小
- B. 从0到声明的大小减1
- C. 任意整数

D. 从0到声明的大小

答案: B

---

## 二、程序填空题（每空5分，共30分）

### 1. 逆序输出数组

```
#include <stdio.h>
int main() {
    int arr[5] = {10, 20, 30, 40, 50};
    for(int i = _____; i >= 0; _____) { // 填空1和2
        printf("%d ", arr[i]);
    }
    return 0;
}
```

答案:

- 填空1: 4
- 填空2: i--

### 2. 求数组元素平均值

```
#include <stdio.h>
int main() {
    float nums[4] = {2.5, 3.5, 4.0, 6.0};
    float sum = 0;
    for(int i = 0; i < _____; i++) { // 填空1
        sum += _____; // 填空2
    }
    printf("平均值: %.2f", sum / 4);
    return 0;
}
```

答案:

- 填空1: 4
  - 填空2: nums[i]
-

### 三、编程题（50分）

#### 1. 查找最大值（20分）

- 定义一个包含6个整数的数组。
- 编写程序找出数组中的最大值并输出。

- 示例：

```
int arr[6] = {3, 9, 2, 7, 1, 5};  
// 输出：最大值是9
```

#### 参考代码：

```
#include <stdio.h>  
int main() {  
    int arr[6] = {3, 9, 2, 7, 1, 5};  
    int max = arr[0];  
    for(int i = 1; i < 6; i++) {  
        if(arr[i] > max) max = arr[i];  
    }  
    printf("最大值是%d", max);  
    return 0;  
}
```

#### 2. 选择排序（30分）

编写一个C语言程序，实现选择排序算法对一维数组进行升序排序，并输出排序后的结果

- 输入一个n值( $1 \leq n \leq 10$ )
- 输入n个整数，用空格隔开
- 输出排序后的结果，用空格隔开
- 输入样例

6

8 3 6 2 7 5

- 输出样例

2 3 5 6 7 8

```
#include <stdio.h>

void selectionSort(int arr[], int size) {
    for (int i = 0; i < size - 1; i++) {
        int minIndex = i;
        // 找到未排序部分的最小值索引
        for (int j = i + 1; j < size; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        // 交换最小值与当前未排序部分的第一个元素
        int temp = arr[i];
        arr[i] = arr[minIndex];
        arr[minIndex] = temp;
    }
}

int main() {
    int arr[] = {8, 3, 6, 2, 7, 5};
    int size = sizeof(arr) / sizeof(arr[0]);

    // 调用选择排序函数
    selectionSort(arr, size);

    // 输出排序结果
    printf("排序结果: ");
    for (int i = 0; i < size; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```