

武汉大学计算机学院

2023-2024 学年度第 1 学期 2023 级

《程序设计 D》期中考试试卷 A 卷 参考答案

姓名：_____ 学号：_____ 班级：_____

说明：答案请全部写在答题纸上，写在试卷上无效。

未经主考教师同意，考试试卷、答题纸、草稿纸均不得带离考场，否则视为违规。

题号	一	二	三	总分
总分	20	20	60	100

一. 多选题：（共 5 小题，共 20 分）

- 下面哪个函数重载的代码是正确的？
 - `int print(int x);
int print(int y);`
 - `int print(int x, int y);
double print(int x, int y);`
 - `int print(int x);
double print(double x);
int print(int x, int y);`
 - `int print(int x);
int print(int x, double y);`
- 以下哪些标识符是不合法的？
 - `ssh.d`
 - `78uf`
 - `_2rr`
 - `re re`
- 设变量 `int m1,m2,n1,n2,a,b` 的值均为 0，则执行表达式 `(m1=a>b)&&(n1=a>=b)` 及表达式 `(m2=a>=b)|| (n2=a>=b)` 后，以下哪几个变量的值为 0？
 - `m1`
 - `n1`
 - `m2`
 - `n2`
- 根据以下程序选择正确的选项：

```
#include<stdio.h>
void main()
{
    int k=_;
    switch(k) {
        case 1:printf("%d",k++);
            break;
        case 2:printf("%d",k++);
        case 3:printf("%d",k++);
```

```

        break;
    case 4:printf("%d",k++);
    default:printf("Full!");
    }
}

```

- A.k=2,程序运行结果为 23
- B.k=2,程序运行结果为 2
- C.k=2,程序运行结果为 34
- D.k=4,程序运行结果为 4

5. 以下选项中，哪个选项中的 printName 函数是内联函数？

A.

```

class Person{
public:
    Person(const string &name) {      Name = name;  }
    void printName(){      cout << Name << endl;    }
private:
    string Name;
};

```

B.

```

class Person{
public:
    Person(const string &name) {      Name = name;  }
    inline void printName();
private:
    string Name;
};
void Person::printName(){      cout << Name << endl;}

```

C.

```

class Person{
public:
    Person(const string &name) {      Name = name;  }
    void printName();
private:
    string Name;
};
inline void Person::printName(){      cout << Name << endl;}

```

D.

```

class Person{
public:
    Person(const string &name) {      Name = name;  }
    void printName();

```

```
private:
    string Name;
};
void Person::printName(){    cout << Name << endl;}
```

【参考答案】

1. C D
2. A B D
3. A B D
4. A C
5. A B C

二. 填空题：（共 5 小题，共 20 分）

6. float a=100/3; 这条语句的执行将导致 a=_____。 int a=100f/3; 这条语句的执行将导致 a=_____。

7. 函数定义如下：

```
double fun(){
    int n;
    .....
    return n;
}
```

那么该函数的返回值是 _____ 类型。

8. 程序如下：

```
...
void f() { ... }
void f(int x = 5, int y = 7) { ... }
int main() {
    ...
    f();
    ...
}
```

编译时报错，原因是_____。

9. 程序如下：

```
struct Student {
    ... //一个较大的结构体，包含学号、姓名、性别、出生日期等各种
    信息
};
void fun(const Student s) {
```

... //对 s 进行处理、输出等

}

如果把 fun() 函数的参数改写为 const Student &s, 该函数的执行效率变得更高, 这是因为 _____。

10. 程序如下:

程序如下:

```
#include <iostream>
class C {
public:
    C() : x(2) { x = 4; }
    int getX() { return x; }
private:
    int x = 0;
};
int main() {
    C obj;
    std::cout << obj.getX();
    return 0;
}
```

程序输出_____。

【参考答案】

6. a=33.0, a=33

7. double

8. 有默认形参的函数重载, 调用时发生二义性

9.. 函数参数传引用时, 形参不用复制实参副本, 节省了大结构体数据复制的时间和空间

10. 输出 4

三. 编程题: (共 3 小题, 每小题各 20 分, 共 60 分)

11. 从键盘依次输入两组整数, 每组不超过 20 个整数, 每组整数中元素不重复, 合并这两组整数, 并使重复的整数只出现一次, 合并后的这组整数按从小到大的顺序排序输出。示例如下:

输入的两组整数为:

5 105 4 32 8 7 9 -60

5 2 87 10 105

输出的结果为:

-60 2 4 5 7 8 9 10 32 87 105

【参考答案】

```
#include<stdio.h>
```

```

int main() {
    int num1, num2, num;
    int temp, i, j;
    int a[20] = { 0 }, b[20] = { 0 }, c[40] = { 0 };

    printf("输入两组数据个数: ");
    scanf_s("%d %d", &num1, &num2);
    for (i = 0; i < num1; i++)
    {
        printf("输入第一组 第 %d 个数据 \n", i);
        scanf_s("%d", &a[i]);
    }
    for (i = 0; i < num2; i++)
    {
        printf("输入第二组 第 %d 个数据 \n", i);
        scanf_s("%d", &b[i]);
    }

    num = num1 + num2;
    for (i = 0; i < num1; i++)
        c[i] = a[i];
    for (i = num1; i < num1 + num2; i++)
        c[i] = b[i - num1];

    printf("合并结果为: \n");
    for (i = 1; i < num; i++)
    {
        temp = c[i];
        for (j = i - 1; j >= 0 && c[j] > temp; j--)
            c[j + 1] = c[j];
        c[j + 1] = temp;
    }

    printf("%d ", c[0]);
    for (i = 1; i < num; i++) {
        if (c[i] == c[i - 1])
            continue;
        else
            printf("%d ", c[i]);
    }
    return 0;
}

```

12. 从键盘输入一组无序的整数(个数少于 20),编写程序按照不同整数出现次数,由多到少输出每个整数及其出现次数,若有整数出现次数相同,则按照输入先后顺序输出这些整数。示例如下:

输入的整数为: 0 -50 0 632 5813 -50 9 -50 0 632

输出的结果:

0: 3

-50: 3

632: 2

5813: 1

9: 1

【参考答案】

```
#include <stdio.h>
```

```
struct Counter {
```

```
    int data;
```

```
    int num;
```

```
};
```

```
int main()
```

```
{
```

```
    struct Counter arr[100], tmpArr;
```

```
    int i, j, m, n, temp, flag;
```

```
    printf("请输入整数的个数: ");
```

```
    scanf_s("%d", &n);
```

```
    m = 0;
```

```
    for (i = 0; i < n; i++)
```

```
    {
```

```
        printf("\n 请依次输入第%d 个整数: ", i);
```

```
        scanf_s("%d", &temp);
```

```
        for (j = 0; j < m; j++)
```

```
            if (temp == arr[j].data)
```

```
            {
```

```
                arr[j].num++;
```

```
                break;
```

```
            }
```

```
        if (j == m)
```

```
        {
```

```
            arr[m].data = temp;
```

```
            arr[m].num = 1;
```

```

        m++;
    }
}

flag = 1;
i = m - 1;
while (i > 0 && flag)
{
    flag = 0;
    for (j = 0; j < i; j++)
        if (arr[j].num < arr[j + 1].num)
        {
            tmpArr = arr[j];
            arr[j] = arr[j + 1];
            arr[j + 1] = tmpArr;
            flag = 1;
        }
    i--;
}

printf("统计的结果为: \n");
for (i = 0; i < m; i++)
    printf("%d: %d\n", arr[i].data, arr[i].num);

return 0;
}

```

13. 某演讲比赛评分时，为了体现公平，在评委给出分数后统计平均得分时，需要去掉最高分和最低分。编写程序，读入评委打分（分数都是大于 0 小于 100 的整数），评委人数大于等于 5，小于等于 50，去掉选手两个最高分和两个最低分，计算并输出选手的平均得分（小数点后保留两位有效数字）。

【参考答案】

```

#include<stdio.h>
int main()
{
    int score[50], i, j, num, max, temp;
    double ave;

    printf("输入评委打分, -1 结束.\n");
    num = 0;

```

```

while (1)
{
    scanf_s("%d", &temp);
    if (temp == -1)
        break;
    else
        score[num++] = temp;
}

for (i = 0; i < num; i++)
{
    max = i;
    for (j = i + 1; j < num; j++)
        if (score[max] < score[j])
            max = j;
    if (max != i)
    {
        temp = score[i];
        score[i] = score[max];
        score[max] = temp;
    }
}

temp = 0;
for (i = 2; i < num - 2; i++)
    temp += score[i];
ave = (double)temp / (num - 4);
printf("选手平均分为:  %.2f\n", ave);

return 0;
}

```