

得分

一、单项选择 (本题共 20 分, 每小题 2 分)

1.1 以下选项中, 不能作为函数返回值的是 ()

- A. char
- B. void
- C. int
- D. delete

1.2 执行语句 `char x = 'a'; int y = x/100;` 以后, 变量 y 的值是 ()

- A. 1
- B. -1
- C. 0.97
- D. 0

1.3 下列说法正确的是 ()

- A. 可以直接用二维数组首地址赋值给普通指针变量
- B. 两个相同类型的指针 p, q, 可以进行相加相减
- C. 一个函数形参被声明为 int* 类型, 它的实参可以是 int 型指针, 也可以是 int 型数组
- D. 指针不能指向数组中某个具体元素的地址

1.4 下列选项中错误的是 ()

- A. 指针变量是把内存中另一个数据的地址作为其值的变量
- B. `int * const p;` 定义了常量指针, `const int * p;` 定义了指针常量
- C. 定义指针时可以不对其初始化
- D. `int a[10], *p=a;` 则 p 指向的是数组 a[0] 的地址

1.5 定义一个交换 a 和 b 的函数, 下列关于函数的声明, 正确的是 ()

- A. `void exchange(int a, int b);` ×
- B. `void exchange(int &a, int b);` ×
- C. `void exchange(int *a, int *b);` ×
- D. `void exchange(int a, int *b);` ×

1.6 下列语句正确的是 ()

- A. `int *a={1,2,3};`
- B. `int a[]=[3,4,5];` ×
- C. `int *a=new int[5];`
- D. `char s="abc";` ×

1.7 下列关于变量的表述中, 正确的是 ()

- A. 全局变量只能被当前程序文件的函数或语句使用
- B. 全局变量只能被主函数访问 ×
- C. 局部变量可以被函数外部的语句使用
- D. 静态变量会被自动初始化

1.8 若函数调用的实参为数组名时, 那么传递给形参的实际上是 ()

- A. 数组首元素
- B. 数组所有元素的值
- C. 数组首地址
- D. 数组所有元素的地址

1.9 下列程序的输出结果是 ()

```
#include<iostream>
using namespace std;
void swap(int *p1,int *p2)
{
    int temp;
    temp=*p1+*p1;
    *p1=*p2;
    *p2=temp;
}
int main(){
    int a=9,b=5;
    swap(&a,&b);
    cout<<a<<" "<<b;
    return 0;
}
```

temp=14 temp=18
a=5 b=18
a=18 b=5

- A. 出错
B. 18,5
C. 59
D. 518

1.10 下列循环代码的循环次数是 (B)

```
int x=5;
do{cout<<x<<endl;
  i++;}while(i!=0);
```

- A. 5
B. 无限
C. 0
D. 2

得分

二、程序改错 (本题共 16 分)

2.1 (8分) 有如下 C++ 代码, 该程序可以实现将字符串 s2 连接到 s1 之后, 输出 s1+s2 的内容。请找出程序中的 4 处错误或缺失, 用横线标出错误或缺失所在位置, 并进行改正。

```
#include <iostream>
using namespace std;
int main() {
    char s1[100] = "happy ";
    char s2[100] = "new year!";
    int length = 0;
    int i = 0;
    while (s1[i] != '\0') {
        i++;
    }
    length = i+1;
    i = 0;
    while (s2[i] != '\0') {
        s2[length + i] = s1[i];
        i++;
    }
    cout << s1 << endl;
    return 0;
}
```

(1) while(s1[i] != '\0')

(2) length = i;

(3) s1[length+i] = s2[i];
(4) s1[length+i] = '0';

2.2 (8分) 有如下 C++ 代码, 该程序可以实现将数字字符串转化为整型。请找出程序中的 4 处错误或缺失, 用横线标出错误或缺失所在位置, 并进行改正。

```
#include <iostream>
using namespace std;
int main() {
    char str[] = "20200101";
    int num;
    int i = 0;
    while (str[i] != '\0') {
        ① if ('0' <= str[i] <= '9') {
            ② num += str[i];
            num *= 10;
        }
        i++;
    }
    ③ cout << "string to number: " << num;
    return 0;
}
```

(1) if (str[i] >= '0' && str[i] <= '9')

(2) num += str[i] - '0';

(3) int num = 0;

(4) char str[9] = "20200101";

在 i 后, i++; 在 i 后, cout 前
加上 num = num / 10;

得分

三、读程序写结果 (本题共 20 分)

20

3.1 (5分)

```
#include <iostream>
using namespace std;
int change(int x) {
    if (x == 1) {
        return 1;
    }
    else if (x == 2) {
        return 2;
    }
    else if (x % 2 != 0) {
        return change((x-1)/2);
    }
}
```


草稿区
5 1 3 2 4

i=1
j=2

j=0 j<1

i=3

j=0 j<

5 3 1

i=1
j=0 j

5 2

```
else{
    return change((x-2)/2)+change(x/2);
}
```

```
int main(){
    int x=18;
    cout<<change(x);
    return 0;
}
```

在 Visual Studio 环境下执行结果:

7

3.2 (5分)

```
#include<iostream>
using namespace std;
int main()
```

```
{
    int n=120,d=0;
    while(n>0)
    {
        d++;
        n=(n+1)/2-2;
    }
}
```

```
cout<<d<<endl;
```

```
return 0;
```

在 Visual Studio 环境下执行结果:

5

3.3 (5分)

```
#include <iostream>
using namespace std;
```

```
int main(){
```

```
    int temp;
```

```
    int array[5]= {5,1,3,2,4};
```

```
    for(int i=0;i<5;i++){
```

```
        for(int j=0;j<i-1;j++){
```

```
            if(array[j]<array[j+1]){
```

```
temp=array[j];
array[j]=array[j+1];
array[j+1]=temp;
```

```
}
```

```
for(int i=0;i<5;i++){
    cout<<array[i]<<" ";
}
```

```
return 0;
```

在 Visual Studio 环境下执行结果:

5 3 2 1 4

3.4 (5分)

```
#include<iostream>
using namespace std;
int main()
```

```
{
    int x=2021,i=0,a[1000],sum=0;
    while(x>0)
```

```
{
    a[i]=x%10;
    x=x/10;
    sum=sum+a[i];
    i=i+1;
}
```

```
cout<<"count="<<i<<endl<<"sum="<<sum<<endl;
return 0;
```

在 Visual Studio 环境下执行结果:

count=4
sum=5

得分
24

四、程序填空 (本题共 24 分, 每空 2 分)

4.1 以下函数的功能是对矩阵进行转置操作。请完善该程序。

```
void trans( double matrix[4][6]){
    double new_matrix[6][4]
```

```
    int i,j;
```

```
    for(i = 0; i < 4; i++){
```

```
        for(j = 0; _____(1)_____ ; j++){
```

```

        new_matrix[j][i] = (2);
    }
}

```

(1) $j < 6$ ✓

(2) $matrix[i][j]$

4.2 以下程序判断一个数是否为回文数，请完善该程序。

```

#include <iostream>
using namespace std;
int main() {
    int N;
    cin >> N;
    int temp = N, sum = 0;
    while (N != 0)
    {
        sum = sum * 10;
        sum = sum * 10 + N % 10;
        N = (2);
    }
    if (3) {
        cout << "yes";
    }
    else {
        cout << "no";
    }
    return 0;
}

```

(1) $sum + N \% 10$ ✓

(2) $N / 10$ ✓

(3) $temp == sum$ ✓

4.3 以下程序对 N 的阶乘进行求和。请完善该程序。

```

#include <iostream>
using namespace std;
int factorial(int N) {
    int res = 1;
    for (int i = 1; (1); i++) {
        res = (2);
    }
    return res;
}

int main() {

```

```

int sum = 0;
int N;
cin >> N;
for (int i = 1; i <= N; i++) {
    sum += (3);
}
cout << sum;
return 0;
}

```

(1) $i \leq N$ ✓

(2) $res * i$ ✓

(3) $factorial(i)$

4.4 以下程序可以去除字符串中重复的字符，请完善该程序。

```

#include <iostream>
#include <string.h>
using namespace std;
int main() {
    char s[100]; // 字符串
    cin >> s;
    char result[100]; // 子串
    int len = 0;
    for (int i = 0; i < strlen(s); i++) {
        bool repeat = false;
        for (int j = 0; j < len; j++) {
            if (s[i] == s[j]) { // 有重复
                (1);
                break;
            }
        }
        if (!repeat) { // 无重复
            result[len] = (2);
            (3);
        }
    }
    result[len] = (4);
    cout << result;
    return 0;
}

```

(1) $repeat = \text{true}$

(2) $s[i]$ ✓

(3) $len++$ ✓

(4) $\backslash 0$ ✓

得分

五、程序设计 (本题共 20 分)

10+10=20

5.1 (10 分) 请设计一个函数实现从数组中删除某个元素的功能。该函数删除数组 `array` 中第一个值为 `k` 的元素, 然后将后面所有元素依次向前移动一位, 最后返回数组长度。如果数组中没有该元素, 则不改变数组, 仍返回数组长度。函数的开头已经给出, 只需要写出完整的函数即可。

```
int delete(int array[], int len, int k);
```

```
int delete(int array[], int len, int k)
```

```
{
    for(int i=0; i<len; i++) // 遍历
```

```
{
    if (arr[i] == k) // 找到 k
```

```
{
    for(int j=i; j<len-1; j++)
```

```
{
    arr[j] = arr[j+1]; // 向前移一位
```

```
}
    return len-1; // 返回长度
```

```
}
return len;
```

5.2 (10 分) 请设计一个程序, 将一个 32 位的二进制数转换成补码形式并输出。程序中关键转换函数的开头需声明为 `void exchange(int array[])`, 其中数组 `array[]` 的长度为 32。

```
#include <iostream>
```

```
using namespace std;
```

```
void exchange(int array[])
```

```
{
    if (arr[0] == 0) // 是正数, 直接输出
```

```
{
    for(int i=0; i<32; i++)
```

```
{
    cout << array[i];
```

```
}
    else // 是负数, 符号位按位求反, 末尾+1
```

```
{
    for(int i=1; i<32; i++)
```

```
{
    if (array[i] == 0) // 转成反码
```

```
{
    array[i] = 1;
```

```
}
    if (array[i] == 1)
```

```
{
    array[i] = 0;
```

```
}
}
```

```
int m = 1; // 反码+1
```

```
for(int i=31; i>=0; i--)
```

```
{
    if (array[i] + m == 2)
```

```
{
    array[i] = 0; // 进位, 本位转 0
```

```
    continue;
```

```
}
    if (array[i] + m != 2) // 不进位
```

```
{
    array[i] = array[i] + m; // 不进位
```

```
    m = 0;
```

```
    continue;
```

```
}
}
```

草稿区

补 = 反 + 1

1: 负

0: 正

```
for(int i=0; i<32; i++)
```

```
{
    cout << array[i];
```

```
}
return i;
```

```
int main()
```

```
{
    int arr[32];
```

```
for(int i=0; i<32; i++)
```

```
{
    cin >> arr[i];
```

```
    exchange(arr[32]); // 调用函数
```

```
    return 0;
```