

C++ 复习提纲

第一章

1. C++的源程序是以.cpp 作为后缀的，目标程序一般以.obj 作为后缀的。
2. 简单语句由 ； 结束。

第二章

1. 标识符只能由字母、数字和下划线 3 种字符组成，且第一个字符必须为字母或下划线。

例题：下列变量名合法的是：

A abc B abc! C 5fun D int

2. 注意算术运算符中的 / 和 %

例题：设 int a=9, b=6, c; 执行语句 c=a/b+0.8; 后 c 的值是：

3. <变量> <赋值运算符=> <表达式>，注意 = 和 == 的区别。

4. ++与—的运算

5. 例题：设 a=5，则执行语句 b=a++; 后，a 和 b 的值分别是

A 6,5 B 6,6 C 5,6 D 5,5

6. 例题：从键盘键入一个字符，判断是大写字母、小写字母还是其它字符。

```
#include <iostream>
using namespace std;
int main()
{
    char c;
    cout<<"please input x:"<<endl;
    c=getchar();
    if(c>='a'&& c<='z')
        cout<<c<<"是小写字母"<<endl;
    else if(c>='A'&& c<='Z')
        cout<<c<<"是大写字母"<<endl;
    else
        cout<<c<<"是其它字符"<<endl;
    return 0;
}
```

第三章

1. 条件 if 语句实现选择，运算符① < ② <= ③ > ④ >= ⑤ == ⑥ !=，结果 false 和 true
2. C++提供 3 种逻辑运算符：(1) && 逻辑与 (2) || 逻辑或 (3) ! 逻辑非

例题：设 `int a=2, b=3, c=4`; 则表达式 `a>b&&b<c||b` 的值是：

例题：判断 `char` 型变量 `ch` 是否为大写字母的正确表达式是：

A `'A' <= ch <= 'Z'`

B `(ch >= 'A') & (ch <= 'Z')`

C `(ch >= 'A') && (ch <= 'Z')`

D `('A' <= ch) AND ('Z' >= ch)`

3. 条件运算符(? :)

例题：设 `int x=10, a=0, b=25`; 则条件表达式 `x<2?a+10:b` 值是：

4. 例题：输入一个字符，判别它是否为大写字母，如果是，将它转换成小写字母；如果不是，不转换。然后输出最后得到的字符。

```
#include <iostream>
using namespace std;
int main()
{
    char ch;
    cin >> ch;
    ch = (ch >= 'A' && ch <= 'Z') ? (ch + 32) : ch;
    cout << ch << endl;
    return 0;
}
```

5. 例题：编写程序判断某一年份是否为闰年。闰年的条件是符合下列情形之一：(1)能够被 4 整除但不能被 100 整除。(2)能够被 100 整除同时又能被 400 整除。

```
#include <iostream>
using namespace std;
int main()
{
    int year;
    cout << "please input year:" << endl;
    cin >> year;
    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))
        cout << year << "是闰年！" << endl;
    else
        cout << year << "不是闰年！" << endl;
    return 0;
}
```

6. 课后练习 10: $y = \begin{cases} x(x < 1) \\ 2x - 1(1 \leq x < 10) \\ 3x - 11(x \geq 10) \end{cases}$ if-else 匹配问题的几点注意

```
#include <iostream>
using namespace std;
int main()
{
    int x, y;
    cout << "please input x:" << endl;
    cin >> x;
```

```

    if(x<1)
        y=x;
    else if(x<10)
        y=2*x-1;
    else
        y=3*x-11;
    cout<<"x="<<x<<","y="<<y<<endl;
    return 0;
}

```

7. **while** 循环、**do-while** 循环、**for** 循环的几点注意事项：① **do-while** 语句是先执行循环体语句，然后判断条件，执行次数为 **n** 大于等于 1 次；② 在 **while** 语句中，是先判断条件，然后执行循环体语句，循环体执行次数为 **n** 大于等于 0 次；③ 循环体可以为空语句。
8. **break**：其作用为使流程从循环体内跳出循环体，即提前结束循环，接着执行循环体下面的语句。**break** 语句只能用于循环语句和 **switch** 语句内，不能单独使用或用于其他语句中；**continue**：其作用为结束本次循环，即跳过循环体中下面尚未执行的语句，接着进行下一次是否执行循环的判定。
9. 例题：求 $1+2+3+\dots+100$ 。

```

#include <iostream>
using namespace std;
int main()
{
    int i=1,sum=0;
    while (i<=100)
    {
        sum=sum+i;
        i++;//叫循环变量
    }
    cout<<"sum="<<sum<<endl;
}

```

10. 例题：求 $1*2*3*\dots*n$ 。

```

#include <iostream>
using namespace std;
int main()
{
    int i, sum=1,n;
    cout<<"please input n:"<<endl;
    cin>>n;
    for(i=1;i<=n;i++)
        sum=sum*i;
    cout<<"sum="<<sum<<endl;
}

```

11. 循环嵌套：

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

      *
    * *
  * * *
* * * *
* * * * *

          *
        * *
      * * *
    * * *
  * * *
* * * *

              *
            * *
          * * *
        * * *
      * * *
    * * *
  * * *
* * * *

                  *
                * *
              * * *
            * * *
          * * *
        * * *
      * * *
    * * *
  * * *
* * * *

```

```
"E:\教学\2013-2014第二学期\c语言(公选)\projects\project01\Debug\project01.exe"
1*1=1 1*2=2 1*3=3 1*4=4 1*5=5 1*6=6 1*7=7 1*8=8 1*9=9
2*1=2 2*2=4 2*3=6 2*4=8 2*5=10 2*6=12 2*7=14 2*8=16 2*9=18
3*1=3 3*2=6 3*3=9 3*4=12 3*5=15 3*6=18 3*7=21 3*8=24 3*9=27
4*1=4 4*2=8 4*3=12 4*4=16 4*5=20 4*6=24 4*7=28 4*8=32 4*9=36
5*1=5 5*2=10 5*3=15 5*4=20 5*5=25 5*6=30 5*7=35 5*8=40 5*9=45
6*1=6 6*2=12 6*3=18 6*4=24 6*5=30 6*6=36 6*7=42 6*8=48 6*9=54
7*1=7 7*2=14 7*3=21 7*4=28 7*5=35 7*6=42 7*7=49 7*8=56 7*9=63
8*1=8 8*2=16 8*3=24 8*4=32 8*5=40 8*6=48 8*7=56 8*8=64 8*9=72
9*1=9 9*2=18 9*3=27 9*4=36 9*5=45 9*6=54 9*7=63 9*8=72 9*9=81
Press any key to continue
```

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    cout<<left;
    for ( int i=1; i<=9; i++)
    {
        for (int j=1; j<=9; j++)
        {
            cout<< i <<"*"<<j<<"="<<setw(3)<<i*j;
        }
        cout<<endl;
    }
    return 0;
}
```

12. 例题：找出 100~200 间的全部素数。

```
#include <iostream>
#include <cmath>
#include <iomanip>
using namespace std;
int main()
{
    int m,k,i,n=0;
    bool prime;          //定义布尔变量 prime
    for(m=101;m<=200;m=m+2) //判别 m 是否为素数，由 101 变化到 200，增量 2
    {
        prime=true;      //循环开始时设 prime 为真，即先认为 m 为素数
        k=int(sqrt(m));   //用 k 代表根号 m 的整数部分
        for(i=2;i<=k;i++) //此循环作用是将 m 被 2 至根号 m 除，检查是否能整除
            if(m%i==0)    //如果能整除，表示 m 不是素数
            {
                prime=false; //使 prime 变为假
                break;       //终止执行本循环
            }
        if (prime)          //如果 m 为素数
```

```

        {
            cout<<setw(5)<<m;    //输出素数 m， 字段宽度为 5
                n=n+1;          //n 用来累计输出素数的个数
        }
        if(n%10==0)  cout<<endl; //输出 10 个数后换行
    }//for 循环结束
    cout<<endl;        //最后执行一次换行
    return 0;
}

```

13. 课后习题 2,3,8。

3-9 有 3 个数 a,b,c, 由键盘输入， 输出其中最大的数。

```

#include <iostream>
using namespace std;
int main()
{
    int a,b,c,max;
    cout<<"please input a,b and c:"<<endl;
    cin>>a>>b>>c;
    max=a;
    if(a<b) max=b;
    if(max<c) max=c;
    cout<<"max="<<max<<endl;
    return 0;
}

```

3-18 求 $\sum_{n=1}^{20} n!$ (即求 $1!+2!+\dots+20!$)。

```

#include <iostream>
using namespace std;
int main()
{
    float s=0,t=1;
    int n;
    for(n=1;n<=20;n++)
    {
        t=t*n;
        s=s+t;
    }
    cout<<"1!+2!+...+20!="<<s<<endl;

    return 0;
}

```

3-19 输出所有水仙花数，所谓水仙花数是指一个 3 位数，其各位数字立方和和该数本身相等。例如 $153=1^3+5^3+3^3$

方法一：

```
#include <iostream>
```

```

using namespace std;
int main()
{
    int ge,shi,bai,n;
    cout<<"水仙花数有: "<<endl;
    for(n=100;n<=999;n++)
    {
        bai=n/100;
        shi=n/10-bai*10;
        ge=n%10;
        if(n==bai*bai*bai+shi*shi*shi+ge*ge*ge)
            cout<<n<<" ";
    }
    cout<<endl;

    return 0;
}

```

方法二:

```

#include <iostream>
using namespace std;
int main()
{
    int i,j,k;
    for(i=1;i<=9;i++)
        for(j=0;j<=9;j++)
            for(k=0;k<=9;k++)
                if(i*i*i+j*j*j+k*k*k==i*100+j*10+k)
                    cout<<i*100+j*10+k<<" ";

    cout<<endl;
    return 0;
}

```

3-21 有一分数序列 $\frac{2}{1}, \frac{3}{2}, \frac{5}{3}, \frac{8}{5}, \frac{13}{8}, \dots$ 求出这个数列的前 20 项之和。

```

#include <iostream>
using namespace std;
int main()
{
    int i,t,n=20;
    double a=2.0,b=1.0,s=0.0;
    for(i=1;i<=n;i++)
    {
        s=s+a/b;
        t=a;
        a=t+b;
    }
}

```

```

        b=t;
    }
    cout<<"s="<<s<<endl;

    return 0;
}

```

14. 输入一个正整数，将其逆序输出。

```

int main()
{
    int n;
    cout<<"input n:"<<endl;
    cin>>n;
    while( n != 0 )
    {
        cout<<n%10<<" ";
        n=n/10;
    }
    cout<<endl;
    return 0;
}

```

15. 输入一个整数，计算它各位上数字的和并输出结果，注意输入的整数是任意位数。

```

#include <iostream>
using namespace std;
int main()
{
    int n,t,sum=0;
    cout<<"input n:"<<endl;
    cin>>n;
    while( n != 0 )
    {
        t=n%10;
        sum=sum+t;
        n=n/10;
    }
    cout<<"sum="<<sum<<endl;
    return 0;
}

```

16. 输出 100~1000 之间所有能被 5 或 6 整除的数，每行输出 10 个。

```

#include <iostream>
using namespace std;
int main()
{
    int i,count=0;
    for(i=100;i<=1000;i++)
    {

```

```

        if(i%5==0 || i%6==0)
        {
            cout<<i<<" ";
            count++;
            if(count%10==0)
                cout<<endl;
        }
    }
    return 0;
}

```

17. 输入 **n** 的值，从键盘随机输入 **n** 个数，求这 **n** 个数最小值和最大值。

```

#include <iostream>
using namespace std;
int main()
{
    int i,n,max,min,number;
    cout<<"input n:"<<endl;
    cin>>n;
    cout<<"input a number:"<<endl;
    cin>>number;
    min=number;
    max=number;
    for(i=1;i<n;i++)
    {
        cout<<"input a number:"<<endl;
        cin>>number;
        if(number>max) max=number;
        if(number<min) min=number;
    }
    cout<<"max="<<max<<",min="<<min<<endl;
    return 0;
}

```

18. 随机输入一批正整数，求这些数的和。

```

#include <iostream>
using namespace std;
int main()
{
    int number,sum=0;
    cin>>number;
    while( number != 0 )
    {
        sum=sum+number;
        cin>>number;
    }
}

```



```

        cout<<"sum="<<sum<<endl;
        return 0;
    }

```

19. 从键盘键入一个整数，判断其的奇偶性；判断其正负。

第四章

1. 无论把一个程序划分为多少个程序模块，只能有一个 **main** 函数。程序总是从 **main** 函数开始执行的。

例题：对于一个正常运行的 C++ 程序，下列叙述正确的是

- A 程序的执行总是起止于 **main** 函数
 - B 程序的执行总是从第一个函数开始，结束于 **main** 函数
 - C 程序的执行总是从 **main** 函数开始，在最后一个函数结束
 - D 程序的执行总是从第一个函数开始，在最后一个函数结束
2. 在程序运行过程中，由主函数调用其他函数，其他函数也可以互相调用。
3. 无参函数，调用函数时不必给出参数。
4. 有参函数，在调用函数时，要给出参数。在主调函数和被调用函数之间有数据传递。
5. 实参变量对形参变量的数据传递是“值传递”，即单向传递，只由实参传给形参，而不能由形参传回来给实参。
6. 函数的返回值是通过函数中的 **return** 语句获得的。
7. 函数值的类型。既然函数有返回值，应当在定义函数时指定函数值的类型。
8. 如果函数值的类型和 **return** 语句中表达式的值不一致，则以函数类型为准，即函数类型决定返回值的类型。对数值型数据，可以自动进行类型转换。
9. 函数声明(**declare**)，就是在函数尚在未定义的情况下，事先将该函数的有关信息通知编译系统，以便使编译能正常进行。
10. C++ 允许用同一函数名定义多个函数，这些函数的参数个数和参数类型不同，即有不同的参数列表，但不能只有函数的类型不同而参数的个数和类型相同。

例题：max 函数重载

```

int max(int a,int b,int c)
double max(double a,double b,double c)
long max(long a,long b,long c)

```

11. 常用的数学库函 **pow()**,**abs()**,**exp()**,**#include<cmath>**
12. 局部变量(**local variable**): ① 在一个函数内部定义的变量; ② 在复合语句中定义的变量
13. 全局变量(**global variable**): ① 在函数之外定义的变量; ② 全局变量的有效范围为从定义变量的位置开始到本源文件结束; ③ 全局变量不进行初始化时，默认值为 0

例题：复合语句中定义的变量的作用范围是：

- A 整个源文件
- B 整个函数
- C 整个程序
- D 所定义的复合语句

第五章

1. 一维数组定义时：① 分别对数组元素赋予初值：**int a [10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};** ② 可以只给一部分元素赋值：**int a [10] = {0,1,2,3,4};** ③ 如果想使一个数组中全部元素值为 1，可以写成：**int a [10] = {1, 1, 1, 1, 1, 1, 1, 1, 1, 1};** ④ 在对全部数组元素赋初值时，可以不

指定数组长度: `int a [5] = {1, 2, 3, 4, 5};` 可以写成: `int a [] = {1, 2, 3, 4, 5};`

2. 不能写成: `int a [10] = {1*10};` 不能给数组整体赋初值。

例题: 在 C++ 语言中, 数组名是

A 一个数组元素 B 数组的第一个元素 C 一个地址常量 D 所有数组元素

3. 二维数组定义时: ① `int array[4][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12} };` ② `int array[4][3] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 };` ③ `int array[4][3] = { {1}, {4, 5}, {7, 8, 9}, {10} };` ④ `int array[4][3] = { 1, 2, 3, 4, 5, 6, 7, 8 };` ⑤ `int array[][3] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 };` ⑥ `int array[][][3] = { {1}, {4, 5}, {7, 8, 9}, {10} };`

例题: 设 `int b[][5] = {1, 2, 3, 4, 5, 6, 7};`, 则元素 `b[1][2]` 的值是

A 0 B 2 C 6 D 7

4. 例题: 输入 10 个数, 求这 10 个数的平均值, 并统计出大于平均值的数的个数以及小于平均值的数的个数
5. 输入 `a[0]~a[4]` 的值, 然后按逆序输出。

```
#include <iostream>
using namespace std;
int main()
{
    const int n=5;
    int a[n], i, temp;
    cout<<"input array a:"<<endl;
    for(i=0; i<n; i++)
        cin>>a[i];
    for(i=0; i<n/2; i++)
    {
        temp=a[i];
        a[i]=a[n-i-1];
        a[n-i-1]=temp;
    }
    for(i=0; i<n; i++)
        cout<<a[i]<<" ";
    cout<<endl;

    return 0;
}
```

6. 输出 Fibonacci 数列的前 20 个值, 每行输出 5 个。

```
#include <iostream>
using namespace std;
#include <iomanip>
int main ()
{
    int f[20], i;
    f[0]=1;
    f[1]=1;
    for(i=2; i<20; i++)
```

```

    {
        f[i]=f[i-1]+f[i-2];
        cout<<"f["<<i+1<<"] = "<<f[i]<<endl;
    }

    for(i=0;i<20;i++)
    {
        cout<<setiosflags(ios::left)<<setw(7)<<f[i];
        if((i+1)%5==0)
            cout<<endl;
    }
    return 0;
}

```

7. 输入 10 个数，求出这 10 个数的最大值和最小值以及最大值的下标和最小值的下标

```

#include <iostream>
using namespace std;
int main ()
{
    const int n=5;
    int a[n],i,min,max;
    cout<<"input array a:"<<endl;
    for(i=0;i<n;i++)
        cin>>a[i];
    max=0;
    for(i=1;i<n;i++)
        if(a[i]>a[max])
            max=i;
    cout<<"最大值的下标是: "<<max<<",最大值是: "<<a[max]<<endl;

    min=0;
    for(i=1;i<n;i++)
        if(a[i]<a[min])
            min=i;
    cout<<"最小值的下标是: "<<min<<",最小值是: "<<a[min]<<endl;
    return 0;
}

```

8. 用冒泡排序法对 10 个数排序(按由小到大顺序); 用选择排序法对 10 个数

冒泡排序

```

#include <iostream>
using namespace std;
int main()
{
    int a[6]={9,8,5,4,2,0},i,j,t;

```

```

    for(i=1;i<6;i++)
    {
        for(j=0;j<6-i;j++)
        {
            if(a[j]>a[j+1])
            {
                t=a[j];
                a[j]=a[j+1];
                a[j+1]=t;
            }
        }
    }
    return 0;
}

```

选择排序

```

#include <iostream>
using namespace std;
int main()
{
    int a[6]={9,8,5,4,2,0},i,j,t,k;
    for(i=0;i<6-1;i++)
    {
        k=i;
        for(j=i+1;j<6;j++)
        {
            if(a[k]>a[j]) k=j;
        }
        t=a[k];
        a[k]=a[i];
        a[i]=t;
    }
    for(i=0;i<6;i++)
        cout<<a[i]<<" ";
    cout<<endl;
    return 0;
}

```

第六章

1. 一个变量的地址称为该变量的指针，指针变量的值是地址。
 2. 有两个与指针变量有关的运算符：
 - ① &取地址运算符。
 - ② *指针运算符（或称间接访问运算符）。
 3. &a 为变量 a 的地址，*p 为指针变量 p 所指向的存储单元。
- 例题：若有定义 `int a[5]={1,2,3,4,5},*p;`，则下列赋值语句中正确的是：

- A $p=\&a$
- B $*p=a$
- C $*(a+1)=a$
- D $p=a+1$

4. $*\&a$ 的含义是：先进行 $\&a$ 的运算，得 a 的地址，再进行 $*$ 运算，即 $\&a$ 所指向的变量

例题：有整型数组和指针定义语句 `int a[10], *p;` 则下面对数组元素 `a[4]` 赋值正确的语句是：

- A `*(p+4)=10;`
- B `*(a+4)=10;`
- C `p[4]=10;`
- D `p+4=10`

5. 例 6.2 输入 a 和 b 两个整数，按先大后小的顺序输出 a 和 b

```
#include <iostream>
using namespace std;
int main( )
{
    int *p1,*p2,*p,a,b;
    cin>>a>>b;                //输入两个整数
    p1=&a;                      //使 p1 指向 a
    p2=&b;                      //使 p2 指向 b
    if(a<b)                    //如果 a<b 就使 p1 与 p2 的值交换
    {p=p1;p1=p2;p2=p;}        //将 p1 的指向与 p2 的指向交换
    cout<<" a=" <<a<<" b=" <<b<<endl;
    cout<<" max=" <<*p1<<" min=" <<*p2<<endl;
    return 0;
}
```

- 6. 数组名就是第一个元素的地址，也可以看作是一个指针常量
- 7. 用数组名作函数的参数，数组名代表数组首元素的地址。
- 8. 用数组名作函数的参数，传递的是数组首元素的地址
- 9. 用指针变量作函数形参，同样可以接收从实参传递来的数组首元素的地址(此时，实参是数组名)
- 10. 下标为 i 的元素的值：

- ① $*(a+i)$
- ② $*(pa+i)$
- ③ $a[i]$
- ④ $pa[i]$

例题：有整型数组和指针定义语句 `int a[10], *p;` 则下面对数组元素 `a[4]` 赋值不正确的语句是：

- A `*(p+4)=10;`
- B `*(a+4)=10;`
- C `p[4]=10;`
- D `p+4=10`

例题：已知 `int a[10]={0,1,2,3,4,5,6,7,8,9},*p=a;` 则不能表示数组 a 中元素的式子是：

- A $*a$
- B $*p$
- C a
- D $p[0]$

11. 下标为 i 的元素的地址:

- ① $a+i$
- ② $pa+i$
- ③ $\&a[i]$
- ④ $\&pa[i]$

例题: 若有定义 `int a[10];`, 则 `a` 数组中首元素的地址可以表示为:

- A `a`
- B `a+1`
- C `&a`
- D `&a[1]`