

day01

day01

一、C++介绍

1.C++背景

2.环境准备

二、第一个程序和注释

1.第一个程序

2.注释

3.快捷键

三、数据类型及变量

1.数据类型

2.变量

1.声明和初始化

2.变量的命名规则

3.常量和输入输出

1.常量

2.输入输出

1.输出

2.输入：获取键盘内容

四、条件与循环

1.条件判断

1.1.if语句

1.2.switch

2.运算符操作

3.三元运算符

五、循环

1.while

2.do...while

3.for

4.break和continue

六、命名空间

1.自定义命名空间

2.简化命名空间的用法

七、数组

1.数组的声明和初始化（操作和C语言一样）

一、C++介绍

1.C++背景

- 1 C++ 是一种通用的，面向对象的编程语言，由Bjarne Stroustrup于1979年设计，是C语言的扩展，并提供了一些额外的功能，例如类，继承，默认函数参数等。
- 2 C++在现代起着不可或缺的作用，因为许多现代系统（例如操作系统，web浏览器，数据库等）在其代码库的至少某些部分中都包含C++代码。而且，由于C++的速度，它在性能至关重要的领域中非常有用。

2.环境准备

- GNU:类Unix操作系统的编译标准，操作系统里内置了
- Visual C++
- MinGW(使用)

二、第一个程序和注释

1.第一个程序

```
1 #include <iostream> // 导入输入输出流依赖
2 // main函数作为C++程序的入口、一个程序只能有一个main
3 int main(){
4     // 打印
5     std::cout << "hello C++" << std::endl;
6     return 0; // 一般返回值是0、表示正常结束
7 }
```

2.注释

- 行注释：快捷键 `ctrl+/`
- 块注释：快捷键 `ctrl+shift+/`

3.快捷键

- `ctrl+p` :查看函数参数解释
- `ctrl+d` :快速复制当前行
- `ctrl+shift+方向上/下` :代码换行
- `Fn+Home/End` :光标跳到最左边或者最右边

三、数据类型及变量

1.数据类型

数据类型	声明
整数类型	int
单精度	float
双精度	double
字符	char
字符串	string
布尔	bool

```

1  #include <iostream>
2  /*
3   * 数据类型
4   */
5  int main(){
6      //    内存占8字节
7      int age = 18;    // 整数
8      //    内存占1字节
9      int8_t age2 = 14;
10     char c = 'a';    // 字符
11     float a = 10.5;
12     double b = 20.8;
13     //    在C/C++中将非0看做true,0看做false
14     bool f = true;
15     bool f2 = false;
16     return 0;
17 }

```

```

1  #include <iostream>
2  /*
3   * 数据类型
4   */
5  int main(){
6      /*数据类型占用空间的大小*/
7      std::cout<<"int占的空间:" << sizeof(int) << std::endl;
8      std::cout<<"char占的空间:" << sizeof(char) << std::endl;
9      std::cout<<"float占的空间 " << sizeof(float) << std::endl;
10     std::cout<<"double占的空间:" << sizeof(double) << std::endl;
11     std::cout<<"bool占的空间:" << sizeof(bool) << std::endl;
12     std::cout<<"int8_t占的空间:" << sizeof(int8_t) << std::endl;
13     std::cout<<"int16_t占的空间:" << sizeof(int16_t) << std::endl;
14     return 0;
15 }
16 // 结果:
17 /*
18 int占的空间:4
19 char占的空间:1
20 float占的空间 4
21 double占的空间:8

```

```
22 bool占的空间:1
23 int8_t占的空间:1
24 int16_t占的空间:2
25 */
```

2.变量

1.声明和初始化

```
1  #include <iostream>
2  int main(){
3      // 声明后立即初始化并赋值
4      int age = 18;
5      // 第二种写法, 先声明
6      int score;
7      // 后赋值
8      score = 100;
9      // C++中的赋值手法
10     int a = 3; // C语言写法
11     int b(5); // 构造的方式
12     int c{4}; // C++11的时候推出的,
13     std::cout<<"a="<<a<<std::endl;
14     std::cout<<"b="<<b<<std::endl;
15     std::cout<<"c="<<c<<std::endl;
16     std::cout<<"a: 占用的空间"<< sizeof(a)<<std::endl;
17     return 0;
18 }
19 // 结果:
20 /*
21 a=3
22 b=5
23 c=4
24 a: 占用的空间4
25 */
```

2.变量的命名规则

数字、字母、下划线, 数字不能开头, 不能有特殊字符, 下划线可以开头

3.常量和输入输出

1.常量

```
1  #include <iostream>
2  /*
3   * 常量:
4   *      1. 与变量成相对的关系
5   *      2. 变量的值可以改变, 常量的值禁止改变
6   */
7  int main(){
8
```

```

9     int age = 18;
10    age = 19;
11    // 常量
12    const int Months_of_year = 12;
13    // 报错
14    // Months_of_year = 20;
15    std::cout<<"age="<<age<<std::endl;
16
17    return 0;
18 }
19 // 运行结果: age=19

```

2.输入输出

输入输出需要导入库 `iostream`，输入是 `std::cin`，输出是 `std::cout`

1.输出

```

1  #include <iostream>
2  int main() {
3
4      std::cout<<"你好";
5      // 单纯换行
6      std::cout<<std::endl;
7      std::cout<<"你好"<<std::endl;
8      std::cout<<"你好";
9      // 不换行,换行的话加\n,或者加
10     // std::endl表示输出行结束,刷新缓冲区内容
11     return 0;
12 }
13 //运行结果:
14 你好
15 你好
16 你好

```

2.输入：获取键盘内容

```

1  #include <iostream>
2  int main(){
3
4      std::cout<<"请输入你的年龄:"<<std::endl;
5      int age;
6      // 输入>>指向age
7      std::cin>>age;
8      std::cout<<"您的年龄是:"<<age<<std::endl;
9      return 0;
10 }
11 // 结果:
12 请输入你的年龄:
13 12
14 您的年龄是:12

```

四、条件与循环

1. 条件判断

1.1. if语句

```
1  #include <iostream>
2  int main(){
3      int age = 0;
4      std::cout<<"请输入你的年龄:"<<std::endl;
5      std::cin>>age;
6      if(age >= 18){
7          std::cout<<"可以参加选举"<<std::endl;
8      }else{
9          std::cout<<"在等等"<<std::endl;
10     }
11     return 0;
12 }
13 // 运行结果:
14 请输入你的年龄:
15 10
16 在等等
```

```
1  #include <iostream>
2  int main(){
3      std::cout<<"请输入你的成绩"<<std::endl;
4      int score;
5      std::cin>>score;
6      if (score>=90){
7          std::cout<<"允许完一会游戏"<<std::endl;
8      }else if(score>=80){
9          std::cout<<"允许看一会电视"<<std::endl;
10     }else{
11         std::cout<<"通宵写作业"<<std::endl;
12     }
13     return 0;
14 }
15 // 运行结果:
16 请输入你的成绩
17 90
18 允许完一会游戏
```

1.2. switch

注意: switch只能允许判定 char、int、枚举

- 只能用于完全的等价判断
- 可以简化我们的多个 else_if

```
1  #include <iostream>
2  int main(){
3
```

```

4     std::cout<<"请输入这次的考试成绩:(A,B,C,D)"<<std::endl;
5     char level;
6     std::cin>>level;
7     switch (level) {
8         case 'A':
9             std::cout<<level<<std::endl;
10            break;
11        case 'B':
12            std::cout<<level<<std::endl;
13            break;
14        case 'C':
15            std::cout<<level<<std::endl;
16            break;
17        case 'D':
18            std::cout<<level<<std::endl;
19            break;
20        default:
21            std::cout<<"输入错误"<<std::endl;
22    }
23    return 0;
24 }
25 // 运行结果:
26 请输入这次的考试成绩:(A,B,C,D)
27 A
28 A

```

2.运算符操作

- 关系运算符
- 逻辑运算符 &&、||、!

```

1  #include <iostream>
2  /*关系运算符*/
3  int main(){
4      std::cout << "请输入语文成绩:" << std::endl;
5      int chinese_score = 0;
6      std::cin>>chinese_score;
7      std::cout << "请输入数学成绩:" << std::endl;
8      int math_score = 0;
9      std::cin>>math_score;
10     if (chinese_score >= 95 && math_score >= 95){
11         std::cout<<"可以去清华附中"<<std::endl;
12     } else {
13         std::cout<<"可以去村里的高中"<<std::endl;
14     }
15     return 0;
16 }
17 // 运行结果:
18 请输入语文成绩:
19 75
20 请输入数学成绩:
21 65

```

3.三元运算符

表达式?值1:值2

五、循环

1.while

2.do...while

3.for

4.break和continue

六、命名空间

1.自定义命名空间

存在相同的变量或者函数时，从而引入了命名空间

定义: namespace 空间名 {}

调用: 空间名::对象、::是域操作符

```

1  #include <iostream>
2  namespace Zhangsan{
3      int age = 19;
4  }
5  namespace Lisi{
6      int age = 18;
7  }
8  int age = 40;    // 全局变量
9  int main(){
10     int age = 30;    // 局部变量，程序采用就近原则
11     std::cout << "age局部 = " << age << std::endl;
12     // ::age访问全局变量
13     std::cout << "age全局 = " << ::age << std::endl;
14
15     std::cout << "张三的年龄" << Lisi::age << std::endl;
16     std::cout << "李四的年龄" << Zhangsan::age << std::endl;
17 }
18 // 结果:
19 age局部 = 30
20 age全局 = 40

```



```
21 张三的年龄18
22 李四的年龄19
```

2.简化命名空间的用法

using简化命名空间

```
1  #include <iostream>
2  // 调用命名空间
3  using namespace std;
4  namespace ZS{
5      int age = 18;
6  }
7  namespace LS{
8      int age = 28;
9  }
10 using namespace ZS;
11 int main(){
12     cout<<"你好"<<endl;
13     cout << "张三的年龄" << age << endl;
14     cout << "李四的年龄" << LS::age << endl;
15     return 0;
16 }
17 // 结果:
18 你好
19 张三的年龄18
20 李四的年龄28
```

七、数组

1.数组的声明和初始化（操作和C语言一样）