

C程序设计语言

第8章 OO与C++、Java、C#

孙志岗
sun@hit.edu.cn
<http://sunner.cn>

C++ 中的Hello World!

- 兼容C语言的:

```
#include <stdio.h>
int main()
{
    printf("Hello, world!\n");
    return 0;
}
```
- 更具C++味道的:

```
#include <iostream>
int main()
{
    cout << "Hello, world!\n";
    return 0;
}
```

2004-12-19

OO & C++, Java, C#

2

Java 中的Hello World!

```
public class Hello
{
    public static void main(String[] args)
    {
        System.out.println("Hello, world!");
    }
}
```

2004-12-19

OO & C++, Java, C#

3

C# 中的Hello World!

```
class HelloWorldApp
{
    public static void Main(string[] args)
    {
        System.Console.WriteLine("Hello, world!");
    }
}
```

2004-12-19

OO & C++, Java, C#

4

What is class?

- **Class**
 - 中文名: 类
 - **class**是C++/Java/C#中的关键字, 用来定义类
 - 类是面向对象世界中的“类型”
 - 或者说, 面向对象的世界中, 一切类型都是类
- **Object**
 - 中文名: 对象
 - 对象是面向对象世界中的“变量”
 - 或者说, 面向对象的世界中, 一切变量都是对象
- **OO**
 - **Object-Oriented**
 - 面向对象

2004-12-19

OO & C++, Java, C#

5

面向对象的世界观

- 世界是由一个一个对象构成的
 - 你、我、他、她、它.....
 - 每个对象都有自己的属性和功能
- 对象与对象之间的相互作用使世界富有生气
 - 相互作用通过使用彼此的功能而进行
- 相似的对象可以归入一类
 - 人类、鸟类、植物类、武器类.....
 - 每一类都有该类对象所共同具有的属性和功能
 - 类之间有继承关系
 - 生物类->动物类->人类->.....
 - 子类继承了父类的属性与功能, 并能够加以改造

2004-12-19

OO & C++, Java, C#

6

面向对象的程序观

- 程序设计是一个在计算机中模拟、处理现实世界的过程
- 结构化程序设计思想
 - 抽取世界的“功能”, 形成函数
 - 函数之间的相互调用构成了对世界的模拟和处理
- 面向对象的程序设计思想
 - 让现实中的“类”与程序中的“类”一一对应
 - 用类定义对象 (现实中的“对象”与程序中的“对象”一一对应)
 - 对象之间按现实情况相互作用, 构成程序

2004-12-19

OO & C++, Java, C#

7

一个游戏问题

- 游戏中有若干种玩家扮演的角色
 - 武士、魔法师、骑士
- 每个角色都具有
 - 攻击力、防御力、智力
 - 战斗
- 魔法师和精灵还具有
 - 魔力
- 每种角色出招的方式都不同

2004-12-19

OO & C++, Java, C#

8

用结构化的C来做

```
■ typedef struct
{
    int type;
    int attack;
    int mentality;
    int defense;
} common_role;

■ typedef struct
{
    int type;
    int attack;
    int mentality;
    int defense;
    int magic;
} magic_role;

void WarriorFight1(common_role* r1, common_role* r2);
void WarriorFight2(common_role* r1, magic_role* r2);
void EnchanterFight1(magic_role* r1, magic_role* r2);
void EnchanterFight2(magic_role* r1, common_role* r2);
.....
```

2004-12-19

OO & C++, Java, C#

9

用结构化的C来做

```
■ common_role w, k;
   magic_role e;

WarriorFight1(&w, &k);
WarriorFight2(&w, &e);
EnchanterFight1(&e, &w);
EnchanterFight1(&e, &k);
KnightFight1(&k, &w);
KnightFight2(&k, &e);
```

2004-12-19

OO & C++, Java, C#

10

用面向对象的C++来做

```
■ class role
{
    int attack;
    int mentality;
    int defense;
    virtual void fight(role* pRole){};
};

■ class Enchanter : public role
{
    int magic;
    virtual void fight(role* pRole)
    { //按魔法师的方式出招 };
};
```

2004-12-19

OO & C++, Java, C#

11

用面向对象的C++来做

```
■ class Warrior : public role
{
    virtual void fight(role* pRole)
    { //按武士的方式出招 };
};

■ class Knight : public role
{
    virtual void fight(role* pRole)
    { //按骑士的方式出招 };
};
```

2004-12-19

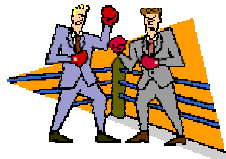
OO & C++, Java, C#

12

用面向对象的C++来做

```
■ Warrior w;  
  Knight k;  
  Enchanter e;
```

```
w.fight(&k);  
w.fight(&e);  
e.fight(&w);  
e.fight(&k);  
k.fight(&w);  
k.fight(&e);
```



2004-12-19

OO & C++, Java, C#

13

用面向对象的C++来做

```
■ role* pRole[3];  
  pRole[0]=&w;  
  pRole[1]=&k;  
  pRole[2]=&e;  
  
for (int i=0; i<3; i++)  
{  
  for (int j=0; i<3; i++)  
  {  
    if (i != j)  
      pRole[i].fight(pRole[j]);  
  }  
}
```

多态

2004-12-19

OO & C++, Java, C#

14

异常 (Exception)

```
■ try  
{  
  open(...);  
  read(...);  
  write(...);  
  close(...);  
}  
catch(Exception &e)  
{  
  //错误处理  
}
```



2004-12-19

OO & C++, Java, C#

15

面向对象之路

- 最重要的在于思考角度的转变
 - 结构化和面向对象并不矛盾
 - 它们的目标相同
 - 让程序可扩展、可复用、易维护.....
 - 它们的基本思想相同
 - 模块化、封装、信息隐藏
- 不要试图从C的角度去理解C++、Java、C#
 - 要认为：它们只是语法上像
 - 忘掉C，然后再看它们
 - 看过它们，再看C

2004-12-19

OO & C++, Java, C#

16

面向对象语言之路

■ C++

- 为了兼容C, 使它有点儿四不像
- 挺复杂, 尤其是加入了STL之后
 - 有人说, STL的泛型思想比OO还要高
- 强大, 需要强人驾驭



■ Java

- 纯面向对象
 - 有时, 似乎在为了对象而对象
- 蕴含很多先进的面向对象模式
- 比较简单, 更适合用来体验OO



■ C#

- 我不懂, ☹。就不妄加评论了

2004-12-19

OO & C++, Java, C#

17

书籍推荐

■ C++编程思想 (第2版) Thinking in C++

- Bruce Eckel
- 卷1, 卷2
- 中文版有卷1

■ Java编程思想 (第3版) Thinking in Java

- Bruce Eckel
- 中文版有第2版

■ 作者提供全部电子版免费下载

2004-12-19

OO & C++, Java, C#

18

书籍推荐

■ 面向对象软件构造 (第2版)

Object-Oriented Software Construction

- Bertrand Meyer
- 购书送英文电子版

■ 设计模式: 可复用面向对象软件的基础

Design Patterns: Elements of Reusable Object-Oriented software

- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
- 中英文版均有

2004-12-19

OO & C++, Java, C#

19