**面向对象与面向过程方法的对比，优缺点以及适用范围。**

面向过程的方法：

优点：

1、从系统整体出发，强调在整体优化的条件下“自上而下”地分析和设计，保证了系统的整体性和目标的一致性；

2、遵循用户至上原则；

3、严格区分系统开发的阶段性，每一阶段的工作成果是下一阶段的依据，便于系统开发的管理和控制；

4、文档规范化，按工程标准建立标准化的文档资料，便于软件在以后的维护。

缺点：

1、重用性差：结构化分析与设计清楚的定义了系统的接口, 当系统对外界接口发生变动时,可能会造成系统结构产生较大变动, 难以扩充新的功能接口；

2、软件可维护性差： 由于软件的可修改性差，导致维护困难，造成维护时费用和成本高，可维护性变差；

3、开发的软件难以满足用户需要：用传统的结构化方法开发大型软件时，往往此系统涉及各种不同领域的知识，在开发需求模糊或需求不断变化的系统时，所开发出的软件系统往往不能真正满足用户的需要。

适用范围：适用于规模较大、结构化程度较高的系统的开发。

面向对象的方法

面向对象开发的缺点：

1、开发过程管理要求高，整个开发过程要经过“修改—评价—再修改”的多次反复；

2、用户过早看到系统原型，误认为系统就是这个模样，易使用户对软件对公司失去信心；

3、开发人员易将原型取代系统分析；缺乏规范化的文档资料，不利于以后的维护

面向对象开发的优点：

1、编程容易。因为面向对象更接近于现实，所以你可以从现实的东西出发，进行适当的抽象；

2、面向对象可以使工程更加模块化，实现更低耦合高内聚思想；

3、符合人们认识事物的规律，系统开发循序渐进，反复修改，确保较好的用户满意度。

面向对象开发的适用范围：

处理过程明确，简单的系统；涉及面窄的小型系统；不适合于大型、复杂系统，存在大量运算、逻辑性强的处理系统

**Java、python、C++各自的优缺点、特点和适用范围。**

**Java**由于其优越的跨平台可移植性，在Web开发中是主流语言。在加上手机Android系统的发展，使得java开发人员的需求量很大。同时，Java现在也用来开发手游。Java 的语法相对规范。

**缺点：**  
不直接支持硬件级别的处理等；使用大量的内存：Java和C++相比使用更多的内存所以占用的内存就更大，启动时间慢。

**Python** 动态解释型，开发效率高，开源，灵活，入门低易上手。

**缺点：**  
Python在移动计算方面是弱的，很少有智能机的应用是Python开发的。python是动态型的语言，需要更多的测试以及错误仅仅是在运行的时候展示的。

**C++**

广泛支持范式的编程语言

帮助更理解其他编程语言，比如说是Java或者是Python是如何进行内存管理以及如何处理缺陷；能调整性能：C++能允许调整应用性能以及影响所有计算机性能。

缺点：

面向过程，分裂比较严重，支持硬件级别的处理

学习困难：C++相对来说学习难度很大

尺寸很大：C++可以称为巨大，它的大部分功能互相影响是机器复杂的方式

**软件过程模型之间的关系**

**1．瀑布模型与RUP模型之间的关系**

在宏观上，瀑布模型是静态模型，RUP模型是动态模型。RUP模型的每一次迭代，实际上都需要执行一次瀑布模型，都要经历先启、细化、构建、产品化这4个阶段，完成瀑布模型的整个过程。

在微观上，瀑布模型与RUP模型都是动态模型。瀑布模型与RUP模型在每一个开发阶段（先启、细化、构建、产品化）的内部，都需要有一个小小的迭代过程，只有进行这样的迭代，开发阶段才能做得更好。

瀑布模型中有RUP模型，反过来，RUP模型中也有瀑布模型。

**2．瀑布模型与增量模型之间的关系**

增量模型是把待开发的软件系统模块化，将每个模块作为一个增量组件，一个模块接着一个模块地进行开发，直到开发完所有的模块。

在开发每个模块时，通常都是采用瀑布模型，从分析、设计、编码和测试这几个阶段进行开发。所以，增量模型中有瀑布模型，即宏观上是增量模型，微观上是瀑布模型。

增量模型也体现了迭代思想，每增加一个模块，就进行一次迭代，执行一次瀑布模型，所以，增量模型本质上是迭代的。

**3．瀑布模型与快速原型模型之间的关系**

快速原型的基本思想是快速建立一个能反映用户主要需求的原型系统，在此基础上之后的每一次迭代，都可能会用到瀑布模型。

快速原型模型中不但包含了迭代模型的思想，而且包含了瀑布模型的思想。

**4．瀑布模型与螺旋模型之间的关系**

螺旋模型是瀑布模型和快速原型模型的结合，快速原型模型是原型模型的简化，原型模型又是迭代模型和瀑布模型的组合，这些模型之间是相互依存的、彼此有关的。

螺旋模型每一次顺时针方向旋转，相当于顺时针方向迭代一次，都是走完一次瀑布模型，这就是瀑布模型与螺旋模型之间的关系。实际上，瀑布模型与喷泉模型也有关系。

**各个模型的适用范围**

1. **瀑布模型：**

结构化方法，面向过程的软件开发方法；

需求变化少；

开发人员熟悉领域；

低风险项目；

使用环境稳定。

**2、原型模型：**

已有产品（原型）；

简单而熟悉的领域；

有快速原型开发工具；

进行产品移植或升级。

1. **增量模型：**

在整个项目开发过程中，需求都可能发生变化，客户接受分阶段交付；

分析设计人员对应用领域不熟悉，难以一步到位；

中等或高风险项目；

用户可参与到整个软件开发过程中；

使用面向对象语言；

软件公司自己有较好的类库、构件库。

1. **螺旋模型：**

庞大、复杂、高风险的系统；

内部开发的大规模软件项目

