



数据结构与算法 (Python版)

队列的应用：打印任务 (上)

陈斌 北京大学 gischen@pku.edu.cn

模拟算法：打印任务

- ❖ 多人共享一台打印机，采取“先到先服务”的队列策略来执行打印任务
- ❖ 在这种设定下，一个首要的问题就是：
这种打印作业系统的容量有多大？
在能够接受的等待时间内，系统能容纳多少用户
以多高频率提交多少打印任务？



模拟算法：打印任务

❖ 一个具体的实例配置如下：

一个实验室，在任意的一个小时内，大约有10名学生在场，

这一小时中，每人会发起2次左右的打印，每次1~20页

❖ 打印机的性能是：

以草稿模式打印的话，每分钟10页，

以正常模式打印的话，打印质量好，但速度下降为每分钟5页。

模拟算法：打印任务

- ❖ 问题是：怎么设定打印机的模式，让大家**都不会等太久**的前提下**尽量提高打印质量**？
- ❖ 这是一个典型的决策支持问题，但无法通过规则直接计算
- ❖ 我们要用一段程序来**模拟**这种打印任务场景，然后对程序运行结果进行**分析**，以支持对打印机模式设定的**决策**。

如何对问题建模？

❖ 首先对问题进行抽象，确定相关的对象和过程

抛弃那些对问题实质没有关系的学生性别、年龄、打印机型号、打印内容、纸张大小等等众多细节



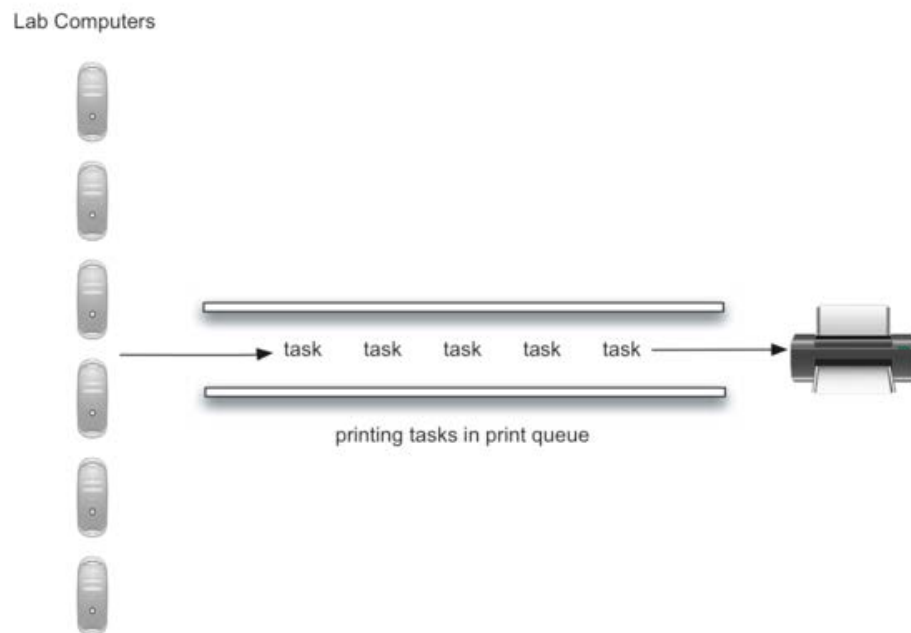
如何对问题建模？

❖ 对象：打印任务、打印队列、打印机

打印任务的属性：提交时间、打印页数

打印队列的属性：具有FIFO性质的打印任务队列

打印机的属性：打印速度、是否忙



如何对问题进行建模？

❖ 过程：生成和提交打印任务

确定生成概率：实例为每小时会有10个学生提交的20个作业，这样，概率是每180秒会有1个作业生成并提交，概率为每秒1/180。

确定打印页数：实例是1~20页，那么就是1~20页之间概率相同。

$$\frac{20 \text{ tasks}}{1 \text{ hour}} \times \frac{1 \text{ hour}}{60 \text{ minutes}} \times \frac{1 \text{ minute}}{60 \text{ seconds}} = \frac{1 \text{ task}}{180 \text{ seconds}}$$

如何对问题进行建模?

❖ 过程：实施打印

当前的打印作业：正在打印的作业

打印结束倒计时：新作业开始打印时开始倒计时，回0表示打印完毕，可以处理下一个作业

❖ 模拟时间：

统一的时间框架：以最小单位（秒）均匀流逝的时间，设定结束时间

同步所有过程：在一个时间单位里，对生成打印任务和实施打印两个过程各处理一次

