

附件 2：案例撰写参考模版

案例名称

一、基本信息

课程名称：Linux 操作系统与程序设计

课程类型：通识教育课 公共基础课 专业课

创新创业课程 实验课

开课年级：大三

面向专业：计算机科学技术、软件工程、网络工程、物联网工程

教学章节：《Beginning Linux® Programming》4th Edition

第 12 章“POSIX Threads”

授课学时：2

主讲教师：阮越

授课形式：MOOC+翻转课堂

选用平台及课程链接：

<https://mooc1.chaoxing.com/mycourse/teachercourse?moocId=8080>

3791&clazzid=12985258

二、案例背景

简要介绍课程性质、课程标准、教学内容体系、学生特点与教学条件等。

课程性质：

“Linux 操作系统与程序设计”是为安徽工业大学计算机科学与技术学院四个专业（计算机科学与技术、软件工程、网络工程、物联网工程）在大三下开设的一门核心专业课程，同时也是一门双语课程。

课程标准：

1) 设计思路与课程目标

学生在学习本课程之前的一个学期,已经学习了操作系统、计算机组成原理、数据库等计算机学科的主干课程。同时在学校两年半的时间内,还学习了 C、C++、Java、Python 等主流程程序设计语言。我们希望通过本课程的学习,一方面使学生可以对操作系统、体系结构中的抽象概念、原理、方法和技术有更深入的理解;另一方面更重要的是,学生能熟悉 Linux 平台上的编程规范和编程技巧,养成在 Linux 平台上工作的习惯和兴趣;通过符合“两性一度”(高阶性、创新性、挑战度)的情境教学,提升学生解决复杂工程问题的能力;借助课程思政,培养学生的综合素养,最终具备在 UNIX/Linux 下开发高质量软件的能力。

2) 课程内容与教材

课程内容主要包括三大块。一是, Linux 的介绍、安装及基本命令(使用)等。相对于 Windows 系统, Linux 有一个门槛,使用系统需要学生记忆较多数量的命令及命令组合,这部分内容的学习也将为后两部分内容打下基础。二是, Linux shell。UNIX/Linux 环境中大量的系统管理程序都是通过 shell 程序完成的,熟悉 shell 可以更好地掌控 Linux 系统。三是, Linux 下的 C 编程。主要包括通过 Linux 系统调用实现文件及文件目录操作,进程/线程的创建、同步、通信等,是这门课的重点和核心。

三块内容中的第二部分(20%内容占比)和第三部分(70%内容占比)使用了 Neil Matthew, Richard Stones 编写的经典英文教材“Beginning Linux Programming (4th Edition)”,第一部分内容由授课教师补充。

3) 教学评价

我校的“计算机科学与技术”专业已经获得国家工程认证,“软件工程”专业等也在积极申报工程认证,“Linux 操作系统与程序设计”是重要的支撑课程。根据指标点分解,本门课程主要帮助确立学生的三个能力训练。

3.1) 工程知识

通过阅读经典代码片段,理解 Linux 下进程、线程等概念,理解文件创建、内存映射、文件目录等,掌握进程/线程调度、进程/线程同步与通信、文件读写的方法。

3.2) 设计/开发解决方案

针对计算机应用领域的复杂工程问题,分析其情境中的具体问题,确定所需用到的 Linux 编程技术,设计相关算法及功能模块;并通过与典型案例的相关代码的分析比较,确定方案的合理性和有效性。

3.3) 研究

能够基于 Linux 平台,设计能实现一定功能,解决具体问题的应用程序。并能根据复杂应用场景的特征,调整方案(算法)参数,或者选择方案中不同的替换技术,最终优化整体方案。

该课程的考核评价基于这三个指标点,并由期末闭卷笔试成绩(50%)和平时成绩(50%)组成。在期末闭卷考试中,涉及 3.1 工程知识的题目分值占比大约 50%, 3.2 和 3.3 各占 25%。平时成绩由学生作业、测验、问题讨论、实验等组成,并且基于超星“一平三端”在线教学平台开展。

4) 教学方法

我们在 MOOC/SPOC 的混合式教学中,基于问题驱动教学法 (Problem-Based Learning, PBL)提炼和总结了二大类共五种问题设计思路。其中一类方法用于强化重要知识点的理解,另一类延展性问题,适用于“高阶性、创新性、挑战度”

的课程建设要求。在后面的案例中，我们会详细介绍我们的教学方法。

教学内容体系：

- 1) Linux 基本命令(2 学时)
 - 1.1) 文件操作类命令 (*)
 - 1.2) 进程控制类命令 (*)
 - 1.3) 系统管理类命令 (Δ)
- 2) Shell 程序设计(4 学时)
 - 2.1) Shell 变量
 - 2.2) 分支与循环 (*)
 - 2.3) 参数
 - 2.4) 函数 (*)
 - 2.5) Sed 与 Awk (Δ)
- 3) Linux C 开发环境(2 学时)
 - 3.1) Vi 编辑器 (*)
 - 3.2) Gcc 编译器 (*)
 - 3.3) Make 工具 (*)
 - 3.4) Gdb 调试工具 (Δ)
 - 3.5) Emacs (Δ)
- 4) Linux 文件操作 (4 学时)
 - 3.1) C 中的基本文件操作函数 (*)
 - 3.2) Linux 文件操作系统调用 (*)
 - 3.3) 内存映射文件
 - 3.4) 文件目录操作
- 5) 进程与信号 (6 学时)
 - 5.1) 创建进程
 - 5.2) 进程同步 (*)
 - 5.3) 替换进程映像
 - 5.4) 信号 (signal) 处理 (*)
- 6) 线程 (4 学时)
 - 6.1) 线程的创建
 - 6.2) 基于信号量机制的线程同步 (*)
 - 6.3) 线程属性 (Δ)
- 7) 进程管道通信 (4 学时)
 - 7.1) 无名管道 (*)
 - 7.2) 有名管道 (Δ)
- 8) IPC (6 学时)
 - 8.1) 消息队列 (*)
 - 8.2) 信号量 (*)
 - 8.3) 共享内存 (*)
- 9) Socket (4 学时)
 - 9.1) TCP (*)
 - 9.2) UDP

注：“*”标示核心知识点 “Δ”标示扩展性知识点

学生特点与教学条件:

年轻学生精力充沛,充满活力,一旦激发出学习热情,往往可以展现出极大的创造力,给教师带来惊喜。

我们在16年开始建设这门课的MOOC资源,18年在安徽省慕课平台“e会学”上线,19年将课程整体迁移到超星平台,并被超星公司认定为“示范课程”。课程资源丰富,除了MOOC视频,还包括电子书、PPT、试题库等其他资源,学生可以利用这些资源在线学习。

三、案例设计思路

授课教师是如何根据本章节教学内容及教学目标确定开展教学的;拟解决的主要问题;如何结合课程内容有机融入疫情素材加强思想政治教育;所采取的教学方法和载体途径。

1 教学思路

本章的视频看完后,学生已经可以在Linux平台利用相关系统调用(函数)建立进程或线程的并发程序。但进程和线程在并发执行的效率和方式上有很大不同,所以如何理解(区分)这种不同是一个关键的点;在并发的进程/线程中,如何利用信号量同步是另外一个重要的问题;线程的创建以及在进程的信号处理中,大量使用了函数指针,虽然在低年级C语言教学中介绍过函数指针,但是同学们并没有真正掌握它的用法,有必要进一步探究和总结它的用法。

2 主要问题

双语教学,给出英文问题:

Question 1:

What is the difference between process and thread? Theory and implementation in Linux?

Please design two programs (utilize the technique of process/thread to achieve the same goal) to illustrate such a difference.

Question 2:

How to synchronize the concurrence of multiple threads? Please present codes to illustrate it.

Question 3:

When invoking `pthread_create`(create a thread), one parameter of this function is

a function pointer. Please describe the role of a function pointer and present code to handle more complicated scenarios by this technique. You can search the internet for the application scenario and also can refer codes on the internet either.

3 思政

课程思政主要基于第 3 个问题---函数指针的总结和提炼展开。

在这次应对新冠肺炎的战役中，火神山医院、雷神山医院起到了定海神针的作用，其中 5G 网络以及稳定的信息系统功不可没。而在底层的通信系统和信息系统的实现中，函数指针有着大量应用。课堂上通过相关报道的介绍，一方面可以令同学们生出国家民族的自豪感；另一方面，也能激发同学的学习热情，树立学成后报效国家的雄心壮志。

4 教学方法

我们使用的是 MOOC+翻转课堂的教学模式。同学们提前看完了对应教材第 12 章“POSIX Threads”的视频后，按照要求以小组为单位提前准备了 3 个问题，然后在在线课堂上，教师随机抽取每个小组（每个小组派一名发言人）回答问题。其他小组点评，或提交更优的方案（问题答案），最后教师总结点评。

四、教学目标

1. 知识与能力目标

这三个问题的前两个在视频课程中都有讲述，难度不大。程度较好的学生在看完视频后，可以比较好地给出复述。类似问题主要用于强化课程中重要知识的理解。

最后一个问题是我们总结的三类延展性问题中的其中一类，即“延展到其他课程情境”中，用于满足“高阶性、创新性、挑战度”的课程建设要求。它主要是希望学生可以讨论实现“函数指针”的高阶应用，实现面向对象程序设计中“类”的基本功能，并和 C++、Java 等语言中类的示例代码比较，理解“重要的是面向对象的思想，而不是面向对象语言本身”。

函数指针是 C 中相对高阶的内容，其中涉及到的知识在课本上要么没有提及，要么只言片语带过，要想比较好地回答这类问题，仅仅看视频（课本）是远远不够的。学生需要以小组为单位，查找文献，给出方案，然后再经过课堂讨论和老师的总结点评，优化方案。在这一过程中，不仅培养了学生独立学习和思考的能力，训练了发散思维和创新思维，也培养了团队合作能力，为学生以后从事研究工作打下基础。

2. 育人目标

通过介绍火神山、雷神山建设和使用中，5G 等信息技术的作用，一方面让同学们为祖国的强大感到骄傲；另一方面，让同学们树立远大理想，学习“匠人

精神”，写出高质量的代码，将来为国家和民族努力奋斗。

五、教学过程

本部分为课程教学实录，应尽量采用写实的方式描述教学过程中的真实情景，如具体有哪些教学环节、教学过程中教师活动和学生活动，尽量将教学中的关键环节以及教学过程中值得注意和思考的现象和事件描述清楚“我是怎么改革的”。特别强调：教师在教学过程中一定要有问题导向，要紧紧围绕所要实现的教学目标，采取灵活多样的组织形式和教学方法手段。

一门学科的课程体系中主干课程间往往有较强的关联性，有一些知识点在不同课程中都有涉及，只不过不同课程的侧重点不同。将本门课程的知识点延展到其他课程中，重新换一个视角考虑问题，往往可以得到意想不到的教学效果。

比如，在C语言中，学生学习了函数指针，但绝大多数同学并不知道怎么用。但在Linux程序设计中，却存在函数指针的大量应用。比如signal函数，它带有函数指针做参数，返回的也是函数指针；再比如线程创建函数pthread_create，也带了函数指针作为参数；而操作信号的高阶函数sigaction更是定义了一个结构体，这个结构体中的分量不仅包含普通的数据类型（整形），也包含了操作信号的函数指针，这实际已经体现了面向对象的程序设计思想。

为此，在本次的翻转课堂中我们设计的高阶问题是“给出一些代码实例，利用函数指针实现面向对象的程序设计”（即前述的第3个问题）。另外，视频课中的某些知识点，或者特别重要，构成了课程的基石；或者较为复杂，理解困难。这些内容就是课程的重难点，虽然视频中对此已有详尽的讲解，但因为缺少交互，教师无法把握学生的理解程度，需要在课堂上加以重点关注。由此，我们又为本次翻转课堂设计了前两个问题。

课程教学的大致过程是这样：课前阶段，教师布置辅助阅读材料和思考讨论题，学生观看视频，并以小组为单位完成讨论题的初步方案；课中阶段，教师给出简要前言，明确本节课要理解和掌握的知识点，然后教师随机抽取小组（选派一名同学）介绍他们的方案，其他小组就发言学生的方案展开讨论，教师则引领

整个讨论过程并对学生的发言分析点评；课后阶段，学生根据课堂讨论的结果，优化自己小组的方案，并最终提交方案（学习心得报告）。

在课堂讨论中，同学们的发言很踊跃。尤其对于最后一个问题，同学们表现出极高的热情，有的同学定义了学生类，并且实现了以学生姓名和学号排序的一致函数接口；有的同学实现了操作符“+”和“-”的重载；还有的同学因为在课堂上没有轮到机会发言，课下和老师私聊讨论。总体上看，翻转课堂上学生的学习热情很高，都勇于展示自己；学习效果也很好，学生给出的代码质量很高，远远超出了老师的预期。

六、教学效果与特色创新

用数据或材料说明在线教学的效果，描述课程设计的新颖独特之处及供借鉴和推广的价值。应提供相应的支撑材料予以证明。

根据我校“Linux 操作系统与程序设计”课程的 MOOC/SPOC 在线教学实践，我们总结和提炼了翻转课堂的核心环节——讨论问题两大类共五种问题设计思路。其中一类方法用于强化重要知识点的理解，另一类延展性问题，适用于“高阶性、创新性、挑战度”的课程建设要求。相关材料以形成教学论文投稿（另附）。

七、教学反思

通过选取和学习疫情相关素材、进行教学设计、开展本次教学改革，谈谈自己所思所想和感悟体会。

本门课程基于 MOOC 或 SPOC 的混合式教学已经建设了一段时间了，其中的 MOOC 资源除了在我们课程组内推广使用，也被超星收录为“示范教学包”，在本次疫情的线上教学中发挥了积极作用。

本次疫情可以选取和学习的素材很多，除了上文介绍的火神山、雷神山建设中 5G 技术的应用，疫情管控中的安康码，抗疫资源的高效配送，都离不开信息技术的支持。通过挖掘这些素材内的思政资源，可以在课程内容中很好地弘扬主旋律，培养学生匠人精神，责任担当，助人等核心价值观。

八、教学资源

1. 与本章节教学内容相关、可供学生课外学习参考的书目、文章、网站等。

2. 学生课前、课中或课后提交的与本章节教学内容相关的典型资料。

1. 在超星平台的课程网站上已经包含了与本章节教学内容相关的资源，另外为了锻炼学生独立研究的能力，一些资源（函数指针的高阶用法）需要学生自行在网上查找。（知网、CSDN、GitHub 等）



2 基于篇幅的考虑, 这里仅展示学生设计的一段代码作为与本章节教学内容相关的典型资料。(第三个问题的方案)。

代码: 操作符重载

```
#include <stdio.h>
#include <stdlib.h>
```



```

// 将 pfOperator 类型定义为一个函数指针
typedef int(*pfOperator)(int, int);
int plus(int,int);
int minus(int,int);
pfOperator getOperator(const char oper);
int main()
{
    int x = 10, y = 20, z = 0;
    pfOperator func = NULL;
    func = getOperator('+');
    z = func(x,y); // 调用 func
    printf("%d\n",z);
    func = getOperator('-');
    z = func(x,y); // 同样是调用 func 得到的结果却不一样
    printf("%d\n",z);
    return 0;
}
pfOperator getOperator(const char oper)
{
    switch(oper)
    {
        case '+':
            return &plus;
            break;
        case '-':
            return &minus;
            break;
        default:
            return NULL;
    }
}
int plus(int x,int y)
{
    return x + y;
}
int minus(int x,int y)
{
    return x - y;
}

```