**实践部分：Anolis OS开源开发实践**

1. 开发任务介绍（背景及意义）

龙蜥社区(OpenAnolis)是由企事业单位、高等院校、科研单位、非营利性组织、个人等按照自愿、平等、开源、协作的基础上组成的非盈利性开源社区。龙蜥社区成立于2020年9月，旨在构建一个开源、中立、开放的Linux上游发行版社区及创新平台。短期目标是开发龙蜥操作系统(Anolis OS)作为CentOS替代版，重新构建一个兼容国际Linux主流厂商发行版。中长期目标是探索打造一个面向未来的操作系统，建立统一的开源操作系统生态，孵化创新开源项目，繁荣开源生态。

龙蜥OS（Anolis OS）是开源操作系统社区OpenAnolis龙蜥社区发起的项目，Alibaba Cloud Linux 是社区发行版Anolis OS 的下游发行版之一。本课程包括理论学习、基础实践、社区贡献三部分。团队将在理论部分讲解操作系统基础知识以及Linux系统学习；基础实践部分需要同学们参与开发者社区-学习中心-Alibaba Cloud Linux技术图谱的平台学习和体验，动手参与到任务实操；社区贡献部分主要让同学们以开发者的身份融入到开源社区的协作与贡献。通过本课程同学可以理解掌握国产云操作系统Anolis OS的核心技术，与社区伙伴和开发者以开放的社区协作来构建一个开源、中立、开放的 Linux 发行版，共建一个开放、合作、共赢的操作系统软件生态。

1. 任务培养目标

学习操作系统基础知识框架，掌握操作系统的基础配置与实现，掌握如何参与开源社区、如何从开源社区获得帮助。

1. 拟培养人数

8人左右

1. 团队导师介绍

梁希，阿里云JVM团队Runtime技术负责人。阿里云开源版JDK Alibaba Dragonwell Maintainer。主导Alibaba Dragonwell上的运行时稳定性、性能、安全特性的研发和大规模落地。

王贇，Michael Wang，阿里云高级技术专家，从事Linux内核研发工作十年以上，2017年回国加入阿里以来，先后带队参与集团业务的混和部署、 NUMA开启以及Share化改造 等多个重大项目，是阿里云操作系统团队调度器领域的带头人。

梁超众，阿里云IoT技术专家，智能手机及IoT行业开发背景，原天猫精灵开发专家，现负责AliOS Things操作系统及HaaS开发框架教育体系。HaaS开发框架是物联网设备云端一体Low-Code开发框架，基于阿里云数亿物联网设备接入经验，提供积木式软硬件开发能力，实现低代码快速开发，帮助中小开发者聚焦业务，实现设备安全上云，加速设备创新迭代。

黄震，阿里云IoT技术专家，长期从事于国产嵌入式实时操作系统的设计与开发。2017年加入阿里，负责AliOS Things操作系统内核部分的规划、设计与部分关键模块的开发工作。带领AliOS Things内核团队实现了弹性内核架构，构建了国内RTOS领域领先诊断维测系统等特色技术。

1. 课程计划（包括16周的时间安排、课程具体内容、阶段性培养目标）

* 理论学习部分（30%） 2周

说明：该部分为线上课程+体验，需在开发者社区-学习中心-Alibaba Cloud Linux技术图谱的平台进行登陆学习

1. Linux基础知识；
2. 操作系统学习；
3. 服务器硬件基础；

* 基础课程体验部分（30%） 5周

说明：该部分为线上课程+体验，需在开发者社区-学习中心-Alibaba Cloud Linux技术图谱的平台进行登陆学习；

1. 初阶任务：Linux 用户及权限操作；
2. 初阶任务：Linux系统管理进阶；
3. 初阶任务：Linux文本管理入门；
4. 中阶任务：TCP/IP网络基础知识与编程
5. 中阶任务：Linux企业常用服务配置；
6. 中阶任务：自动化运维：Shell脚本进阶、Python脚本入门；
7. 中阶任务：系统安全：iptables 安全、系统安全及业务安全；
8. 高阶任务：HTTP服务配置与使用：HTTP代理系统、高性能集群负载；
9. 高阶任务：运维监控系统；
10. 高阶任务：自动化运维：运维自动化学习、Python脚本进阶
11. 高阶任务：KVM虚拟化配置与管理；

* 开源贡献部分（40%） 5-7周

说明：该部分需要同学参与社区贡献，包括参与社区运营、文档贡献和代码贡献三方面。

1. 社区运营：

订阅社区邮件列表；

社区官微技术文章编辑；

1. 参加任一个SIG，向社区提交技术文档
2. 撰写参与社区贡献的体验文章；

基于开源社区的动手实践课题，如下题目任选其一：

题目1: 易调度器的设计和实现，一键动态切换调度器

云原生场景下，虚拟机的规格会变的越来越小，传统的CFS调度器在CPU数量很少的Guest内显的非常臃肿，开销大并且大部分功能也是不需要的。

简易调度器项目的目标，是基于小规格虚拟机场景，从零出发构建一个新的调度器，即通过极简设计提升稳定性和性能表现，在确保多任务正常流转的前提下，把调度延迟或吞吐做到极致（或能用开关在两者之间切换）

题目2:基于AliOS Things操作系统提供云文件系统

文件系统是操作系统的重要组成部分，用来管理和存储大量的文件信息，负责对文件的存储空间进行分配和管理，并对存入其中的文件进行保护和检索，同时为用户提供包括文件创建、删除、命名、读写、访问控制等一系列功能。此外，文件系统还可以根据存取权限及访问操作类型来指定用户对文件的存取。

云存储伴随着云计算产生，作为云计算的先驱，它很早就进入了广大研究人员的视野，众多企业都将其作为进军云计算的第一步，目前也有很多企业面向大众提供云盘服务的企业。阿里云也面向公众提供云存储服务。

云盘功能在移动端操作系统中大多以独立应用程序的形式存在，和本地操作系统没有很好的融。

AliOS Things是阿里云智能IoT开发的物联网操作系统，在物联网领域有大量广泛的应用。

本实验的目标是在AliOS Things上面，打通云文件系统的功能。主要目标主要有三点：

1. 帮助学生掌握操作系统提供的文件系统接口的功能及需求

2. 帮助学生掌握常用云存储接口

3. 帮助学生掌握端云一体开发方式的一般流程

题目3:基于AliOS Things操作系统提供线程级读写锁功能

AliOS Things中，互斥量（mutex）实现了最基础的线程（即task）互斥。 此外，业界还存在一种与互斥量类似的实现线程互斥的机制----读写锁。它实现更细粒度的线程互斥，允许更高的并行性。其特性为：

多个读操作可以并行：即同一时刻允许多个读者并行操作。

多个写操作之间互斥：即同一时刻只允许一个写者操作。

读操作与写操作之间互斥：即不允许读者与写者并行。

本实验的目标是在AliOS Things上面实现线程级读写锁功能。

题目4:基于AliOS Things操作系统提供用于管理堆（heap）的动态内存管理算法

AliOS Things中提供k\_mm\_alloc与k\_mm\_free两个用于申请/释放动态内存的API，在某些应用场景下他们面临与C库malloc与free类似的问题：

无法实现使用者对内存块的字节对齐要求

无法实现多内存分区的分离管理（如某些芯片内存资源分为快速SRAM与普通DRAM，使用者期望能指定从SRAM或是DRAM进行申请）

本实验的目标是在AliOS Things上面实现新的堆内存管理算法，满足以上两点外部功能要求。

1. 评分标准（考核方式，评分细则）

理论学习、基础课程体验、社区贡献分别占考核比重的20分、40分、40分。

* 理论学习部分为20分，学习完成并撰写一篇Linux 系统探索类文章（优质文章可推荐至龙蜥官网）。
* 基础实践部分分为初级、中级、高级任务，分别占10分，四个题目，自选一个，为10分。
* 开源贡献部分包括社区运营、贡献文档和贡献代码三部分。其中参与社区运营满分10分、贡献技术文档满分15分，撰写参与社区贡献的体验文章15分

1. 课程资源（团队可以提供的资源，可选）

阿里云开发者社区Alibaba Cloud Linux 技术图谱

（<https://developer.aliyun.com/graph/linux?spm=a2c6h.23029176.J_9880326140.5.6f5d45bbFJGjxc>）

龙蜥 OS 操作指南：系统安装与装配

（https://openanolis.cn/video/378586449572708960）

8、联系方式

题目1: 王贇 钉钉号：u0aoxf3

题目2: 梁超众 钉钉号：ethanlcz

题目3&4: 黄震 钉钉号；jurassicmoon

龙蜥社区Java 语言与虚拟机SIG实践：梁希钉钉号：knlho8e

龙蜥社区DDE SIG实践：王新 ，手机号码17762462550

实践大纲的补充建议：

* **实践内容，需要安排项目架构解读和源代码阅读（提前公布预习内容）**
* 设置项目实践预习内容，让同学们预先了解项目实践可选择的内容，以便后期选择具体的项目实践任务
* 利用1-2节课时间对实践任务做更加清晰的剖析，对同学们后期实践有帮助
* 可以在实践里加入答题、互动加分环节，调动同学们的积极性
* 可以在上述大纲中加入对同学们选课基础的说明（注意如果太严苛有可能导致团队选择的人数过少）
* 可以在上述相对固定的任务和导师团队的基础上，加入自选题环节，导师可以社区内流动化指导，体现开源优势

实践实施说明和建议：

* 实践部分的授课可以采取线上或者线下两种形式，每周固定授课时间在2小时以上，平时尽可能对同学们的问题及时反馈；
* 在课程预选或者初始阶段尽早进行课下接触可以增加团队凝聚力
* 在实践课的时候录屏，以供同学们反复学习