

# 舞蹈机器人基地第三轮考核

前言：按照之前的流程，Ubuntu的基础学习会单独进行一次考核，确定方向后会根据不同方向所用的知识进行分开考核。但今年他们好像把Ubuntu基础学习给省略了，文件上第三轮是ros2的内容。考虑到前两轮考核均未涉及Ubuntu，本次考核主要还是围绕Ubuntu进行，然后涉及一些ros的知识！

## 一、Ubuntu双系统的安装

由于ros2只支持Ubuntu22.04版本，而本次考核涉及ros1、ros2的基础学习，所以呢给大家三个ubuntu版本安装的建议。

- 先安装18.04或者20.04版本并进行ros1的学习，然后卸载重装22.04版本，进行ros2的学习

麻烦但实用，不会出现很多错误，但记得不要卸载错误（某位学姐曾把Windows删除）

- 直接安装22.04版本，因为22.04支持ros1和ros2，但你需要确保能够转换ros1和2时能够删除干净（22.04默认ros是ros2，ros1需要源码编译）

可能会出现很多问题

- 直接安装ubuntu18.04/20.04和22.04两个版本

这需要你的电脑有足够的内存

以上三个方法各有利弊，看自己能力和想法选择合适的方法

**提交要求：提交能够证明自己已经安装系统的图片即可**

## 二、Ubuntu系统的相关配置

### 1. 更换镜像源

所谓工欲善其事必先利其器，更换镜像源是非常重要且必要的一步，由于Ubuntu的官方源在国外，下载软件可能会很慢，所以我们就需要换成国内的镜像源。请合理选择适合自己的镜像源。

### 2.IDE的安装和配置

- **vscode**

VSCode，是微软在2015年4月30日build开发者大会上发布的一个跨平台源代码编辑器。说它是源代码编辑器，可是它又能进行简单的代码调试运行。说它是 IDE，可是他启动速度之快，令其他 IDE 望尘莫及。而且，在座的各位见过哪个 IDE，能把 c 语言，java，python，JavaScript 等语言写个遍的？VSCode 最厉害的点就在这里——结合它的插件市场，加上电脑内的语言环境，VSCode 能支持37种语言或文件（截至2019年9月）。

优点

第一， 开源，免费，跨平台。

第二， 有完善的插件生态，VSCode 的插件功能种类繁多，从代码样式更改到代码提示补全，再到代码运行调试格式化，只要找到相应的插件，就能给自己的 VSCode 添加相应的功能。

第三， 内置了 Emmet 插件

第四， 背靠微软大佬，熟悉程序员需求。微软的另一款产品 visual studio，被戏称为宇宙第一 IDE，

因为他着实好用，**VSCode** 是在**visual studio** 的基础上开发而来的（参考了需求和样式）。

第五， 内置了 **git** 。版本管理工具的重要性只要有参与代码开发就一定知道，**VSCode** 直接内置了 **git**，令版本管理更加方便。

第六， 使用相应语言的插件拓展，可以对代码进行简单的调试。众所周知，编程软件提供的功能对于很多开发者而言只有 20% 比较常使用，而 **VSCode** 结合插件后就能做到那最关键的 20% 的功能，对于对 **IDE** 只是轻量需求的开发者来说这绝对是个福音。（别忘了 **VSCode** 的启动速度和流畅度以及内存占用）。

第七， 语法高亮和智能提示

- **pycharm**

**Pycharm**是由**jetbrains**开发的优秀的**python IDE**。正如所有其它 **JetBrains** 集成开发环境一样，**PyCharm** 具有智能代码编辑器，能理解 **python** 的特性并提供卓越的生产力推进工具：自动代码格式化、代码完成、重构、自动导入和一键代码导航等。这些功能在先进代码分析程序的支持下，使 **PyCharm** 成为 **python** 专业开发人员和刚起步人员使用的有力工具。

目前已更新到**Pycharm 3**，发布两个版本：**Professional Edition** 和**Free Community Edition**。**Professional Edition**是付费版，提供更加高级的扩展功能，而**Free Community Edition**属于免费版，并没有试用时间限制。如果使用者不是非常必要使用付费版本的高级功能，免费版已经可以胜任大部分的工作。

## 三、Ubuntu系统的操作

在以后的操作过程中，我们会逐渐减少在编译器中运行程序，而是在命令行发布命令直接运行。

### 1、以下是命令行的基本操作

- (1) **ctrl c**：取消命令，并且换行
- (2) **ctrl u**：清空本行命令
- (3) **tab**键：可以补全命令和文件名，如果补全不了快速按两下**tab**键，可以显示备选选项
- (4) **ls**：列出当前目录下所有文件，蓝色的是文件夹，白色的是普通文件，绿色的是可执行文件
- (5) **pwd**：显示当前路径
- (6) **cd XXX**：进入XXX目录下，**cd ..** 返回上层目录
- (7) **cp XXX YYY**：将XXX文件复制成YYY，XXX和YYY可以是一个路径，比如../dir\_c/a.txt，表示上层目录下的dir\_c文件夹下的文件a.txt
- (8) **mkdir XXX**：创建目录XXX
- (9) **rm XXX**：删除普通文件；**rm XXX -r**：删除文件夹
- (10) **mv XXX YYY**：将XXX文件移动到YYY，和**cp**命令一样，XXX和YYY可以是一个路径；重命名也是用这个命令
- (11) **touch XXX**：创建一个文件
- (12) **cat XXX**：展示文件XXX中的内容

### 任务要求

请你简单学习命令行相关文件操作，完成以下任务：

在家目录下创建homework文件夹，分别创建文件夹dir\_a, dir\_b, dir\_c,  
进入dir\_a,  
在文件夹下创建a.txt，再复制成：a.txt.bak, b.txt.bak, c.txt.bak  
将a.txt,重命名为：a\_new.txt  
将dir\_a文件夹下的a.txt.bak 移动到文件夹dir\_b下  
将dir\_a下的a\_new.txt 删除  
将文件夹dir\_c删除

将每一步所执行的指令记录，并用tree命令展示最后的文件层级。

## 2、程序的编译

请分别用c++ 和python在命令行中输出“hello world”

提交要求：提交1和2相应操作的截图

# 四、ros的安装、学习与使用

## 1、安装

安装过程可能因为网络的问题，出现 rosdep update 超时现象。但是，只要能打开小乌龟，都无伤大雅。（关于rosdep update超时的处理，在CSDN上有相关的教程与说明）

## 2、新建包并catkin\_make

## 3、学习

完成[ros wiki](#)中的初级教程并记录学习过程

完成[ros 2](#)的学习和了解

了解ros1与ros2的区别并记录

## 4、使用

在学习了ros1和ros2后，分别使用cpp和py编写一个简单的发布节点和订阅节点，发布和接受的内容为自己的学号

提交要求：提交小乌龟截图（证明ros以安装）、学习笔记、发布接受学号截图

# 进阶：有能力可做

## 1、节点互传（分topic和C-S两种方式完成）

**描述：**共 3 个 ROS 节点，其中 a、b 两个节点分别作为输入节点，c 作为输出节点，分别从ab 节点输入两个数并发布，由 c 节点订阅对应消息，计算两个数的加和并打印显示出来，同时将结果发布出去，ab节点订阅对应消息并打印输

出。

注：此任务要求cpp和py均提交

## 2、设计一个电池电量模拟器，具体明细如下：

- 有一个**参数服务器节点** (Service Node) ，其广播一个Service，提供电池电量的下降速率和初始电量等信息。（信息值具体可以写死，也可以终端输入）。
- 有一个**发布者节点** (Publisher Node) ，该节点启动时先向服务器节点调用/索要电池初始值和电池耗电速率相关信息，后按照电池初始值和电池耗电速率，以特定频率发布且更新实时电池电量于Topic。此发布者节点代码文件/脚本内还应有一个充电服务 (Service) ，被调用后可以立马充电到100。（即此Node既要向参数服务器索要起始信息，又要按频率发布Topic，又要广播一个Service来支持被调用以充电。）
- 有一个**订阅者节点** (Subscriber Node) ，其监听电池电量并输出到终端，当电池电量小于阈值，自动调用充电服务。提示：以上三个节点，每个节点为一个代码文件（.cpp和.h配套）。可能需要写.srv文件。均要求使用C++编写。
- 再提示：package.xml 和 CMakeLists.txt 对初学者蛮难写的。

## 以上内容均提交到各自组负责人邮箱，考核时间为两周

写在最后：这本来应该是两次考核的内容，但被安排到了一次。我已经减少了对ros的考核，侧重于Ubuntu系统的一些知识，希望大家可以坚持下去，学长学姐首先看的是你的态度，其次才是你完成的质量。如果在截至时间没有完成，可以先把完成的东西提交了，后面慢慢补。进阶部分只是预防有一些天赋异禀的人，没有完成的同学不用担心。