

# Assignment #1: 虚拟机，Shell & 大语言模型

Updated 1730 GMT+8 Feb 20, 2025

2025 spring, Complied by 郑涵予 物理学院

## 作业的各项评分细则及对应的得分

标准	等级	得分
按时提交	完全按时提交：1分 提交有请假说明：0.5分 未提交：0分	1分
源码、耗时（可选）、解题思路（可选）	提交了4个或更多题目且包含所有必要信息：1分 提交了2个或以上题目但不足4个：0.5分 没有提供源码：0分	1分
AC代码截图	提交了4个或更多题目且包含所有必要信息：1分 提交了2个或以上题目但不足4个：0.5分 没有提供截图：0分	1分
清晰头像、PDF文件、MD/DOC附件	包含清晰的Canvas头像、PDF文件以及MD或DOC格式的附件：1分 缺少上述三项中的任意一项：0.5分 缺失两项或以上：0分	1分
学习总结和个人收获	提交了学习总结和个人收获：1分 未提交学习总结或内容不详：0分	1分
总得分： 5	总分满分： 5分	



- 说明：
- 1. 解题与记录：

- 对于每一个题目，请提供其解题思路（可选），并附上使用Python或C++编写的源代码（确保已在OpenJudge，Codeforces，LeetCode等平台上获得Accepted）。请将这些信息连同显示“Accepted”的截图一起填写到下方的作业模板中。（推荐使用Typora <https://typoraio.cn> 进行编辑，当然你也可以选择Word。）无论题目是否已通过，请标明每个题目大致花费的时间。

## 2. 课程平台与提交安排：

- 我们的课程网站位于Canvas平台（<https://pku.instructure.com>）。该平台将在第2周选课结束后正式启用。在平台启用前，请先完成作业并将作业妥善保存。待Canvas平台激活后，再上传你的作业。
  - 提交时，请首先上传PDF格式的文件，并将.md或.doc格式的文件作为附件上传至右侧的“作业评论”区。确保你的Canvas账户有一个清晰可见的头像，提交的文件为PDF格式，并且“作业评论”区包含上传的.md或.doc附件。

## 3. 延迟提交：

- 如果你预计无法在截止日期前提交作业，请提前告知具体原因。这有助于我们了解情况并可能为你提供适当的延期或其他帮助。

请按照上述指导认真准备和提交作业，以保证顺利完成课程要求。

# 1. 题目

## 27653: Fraction类

<http://cs101.openjudge.cn/practice/27653/>

思路：直接通分计算即可，注意最后分子分母要约分。（用时约5min）

代码：

```
def gcd(a,b):
    if a%b==0:return b
    else:return gcd(b,a%b)
class Fraction:
    def __init__(self,up,down):
        self.up = up
        self.down = down
    def __str__(self):
        return str(self.up)+"/"+str(self.down)
    def __add__(self,other):
        x=self.up*other.down+self.down*other.up
        y=self.down*other.down
```

```

d=gcd(x,y)
return Fraction(x//d,y//d)
a,b,c,d=map(int,input().split())
print((Fraction(a,b)+Fraction(c,d)))

```

代码运行截图 (至少包含有"Accepted")


**CS101 / 题库 (包括计概、数算题目)**

[题目](#)
[排名](#)
[状态](#)
[提问](#)

---

#48367883提交状态

[查看](#)
[提交](#)
[统计](#)
[提问](#)

状态: Accepted

源代码

```

def gcd(a,b):
    if a%b==0: return b
    else: return gcd(b,a%b)
class Fraction:
    def __init__(self,up,down):
        self.up = up
        self.down = down
    def __str__(self):
        return str(self.up)+"/"+str(self.down)
    def __add__(self,other):
        x=self.up*other.down+self.down*other.up
        y=self.down*other.down
        d=gcd(x,y)
        return Fraction(x//d,y//d)
a,b,c,d=map(int,input().split())
print((Fraction(a,b)+Fraction(c,d)))

```

基本信息

# : 48367883  
题目: 27653  
提交人: 2400011325  
内存: 7384kB  
时间: 33ms  
语言: Python3  
提交时间: 2025-02-26 17:05:09

©2002-2022 POJ 京ICP备20010980号-1

[English](#)
[帮助](#)
[关于](#)

## 1760.袋子里最少数目的球

<https://leetcode.cn/problems/minimum-limit-of-balls-in-a-bag/>

思路：遇到最大值最小的问题优先考虑二分。直接验证即可。（用时约5min）

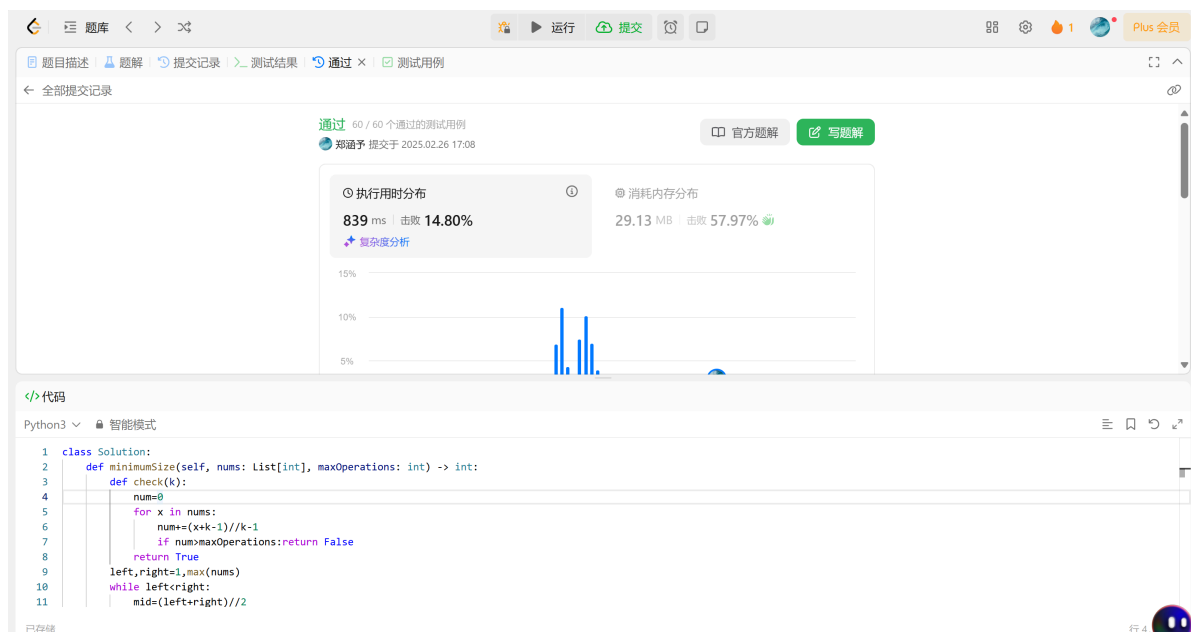
代码：

```

class Solution:
    def minimumSize(self, nums: List[int], maxOperations: int) -> int:
        def check(k):
            num=0
            for x in nums:
                num+=(x+k-1)//k-1
                if num>maxOperations: return False
            return True
        left,right=1,max(nums)
        while left<right:
            mid=(left+right)//2
            if check(mid): right=mid
            else: left=mid+1
        return left

```

## 代码运行截图 (至少包含有"Accepted")



## 04135: 月度开销

<http://cs101.openjudge.cn/practice/04135>

思路：同样考虑二分，思路和上一题类似。但是调bug多用了点时间。(用时约15min)

代码：

```
n,m=map(int,input().split())
a=list(int(input()) for _ in range(n))
def check(x):
    num=1
    cur=0
    for i in range(n):
        if cur+a[i]>x:
            cur=a[i]
            num+=1
        else:cur+=a[i]
        if num>m:return False
    return True
left,right=max(a),sum(a)+1
ans=-1
while left<right:
    mid=(left+right)//2
    if check(mid):
```

```

        right=mid
        ans=mid
    else:left=mid+1
print(ans)

```

代码运行截图 (至少包含有"Accepted")

OpenJudge 题目ID, 标题, 描述 2400011325 信箱 账号

CS101 / 题库 (包括计概、数理题目)

题目 排名 状态 提问

#48368476提交状态 查看 提交 统计 提问

状态: Accepted

源代码

```

n,m=map(int,input().split())
a=list(int(input()) for _ in range(n))
def check(x):
    num=1
    cur=0
    for i in range(n):
        if cur+a[i]>x:
            cur=a[i]
            num+=1
        else:cur+=a[i]
    if num>m:return False
    return True
left,right=max(a),sum(a)+1
ans=-1
while left<right:
    mid=(left+right)//2
    if check(mid):
        right=mid
        ans=mid
    else:left=mid+1
print(ans)

```

基本信息

- #: 48368476
- 题目: 04135
- 提交人: 2400011325
- 内存: 7508kB
- 时间: 601ms
- 语言: Python3
- 提交时间: 2025-02-26 17:33:13

©2002-2022 POJ 京ICP备20010980号-1 English 帮助 关于

## 27300: 模型整理

<http://cs101.openjudge.cn/practice/27300/>

思路：注意到题目中所给的所有模型参数中，单位是"B"的永远比单位是"M"的大，所以不需要转化前面的数字，直接自定义比较函数，先比单位再比数字即可。(用时约10min)

代码：

```

from collections import defaultdict
import functools
def cmp(x,y):
    if x[-1]!=y[-1]:
        return -1 if x[-1]=='M' else 1
    return -1 if float(x[:-1])<float(y[:-1]) else 1
mp=defaultdict(list)
n=int(input())
for _ in range(n):
    s1,s2=input().split('-')
    mp[s1].append(s2)

```

```
for key in sorted(mp.keys()):
    print(key,end=": ")
    mp[key].sort(key=functools.cmp_to_key(cmp))
    print(' ', '.join(mp[key]))
```

代码运行截图 (至少包含有"Accepted")



## Q5. 大语言模型（LLM）部署与测试

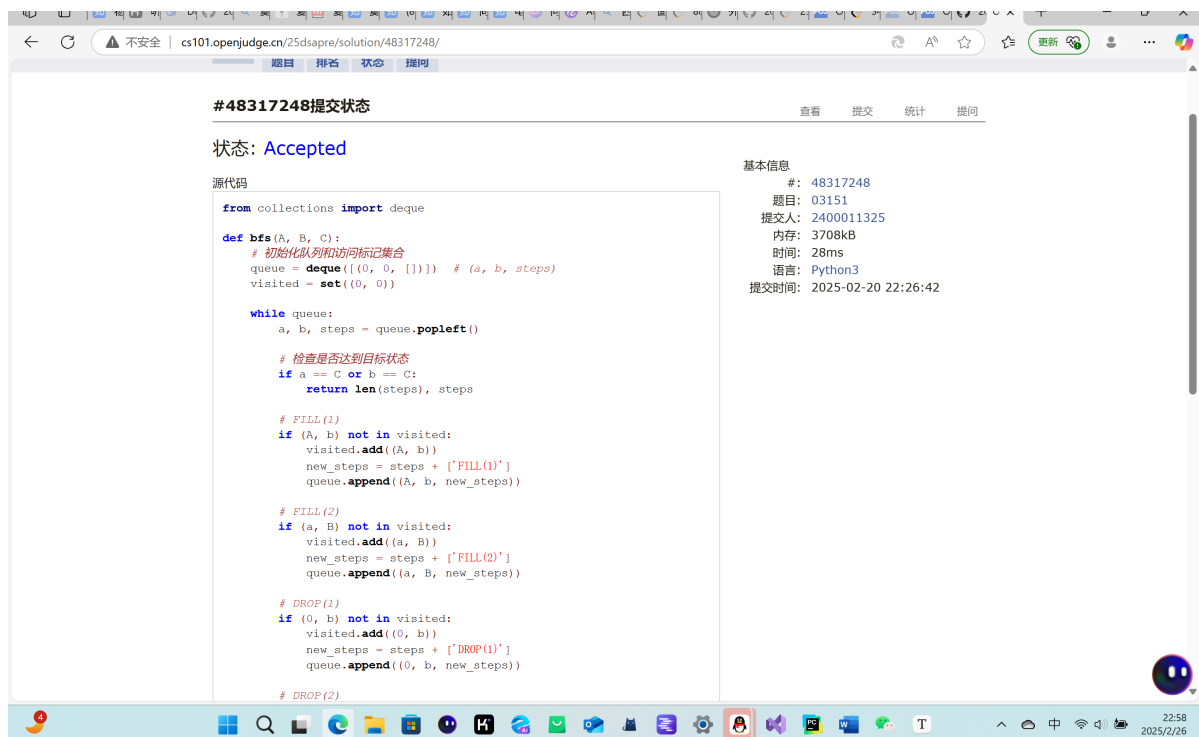
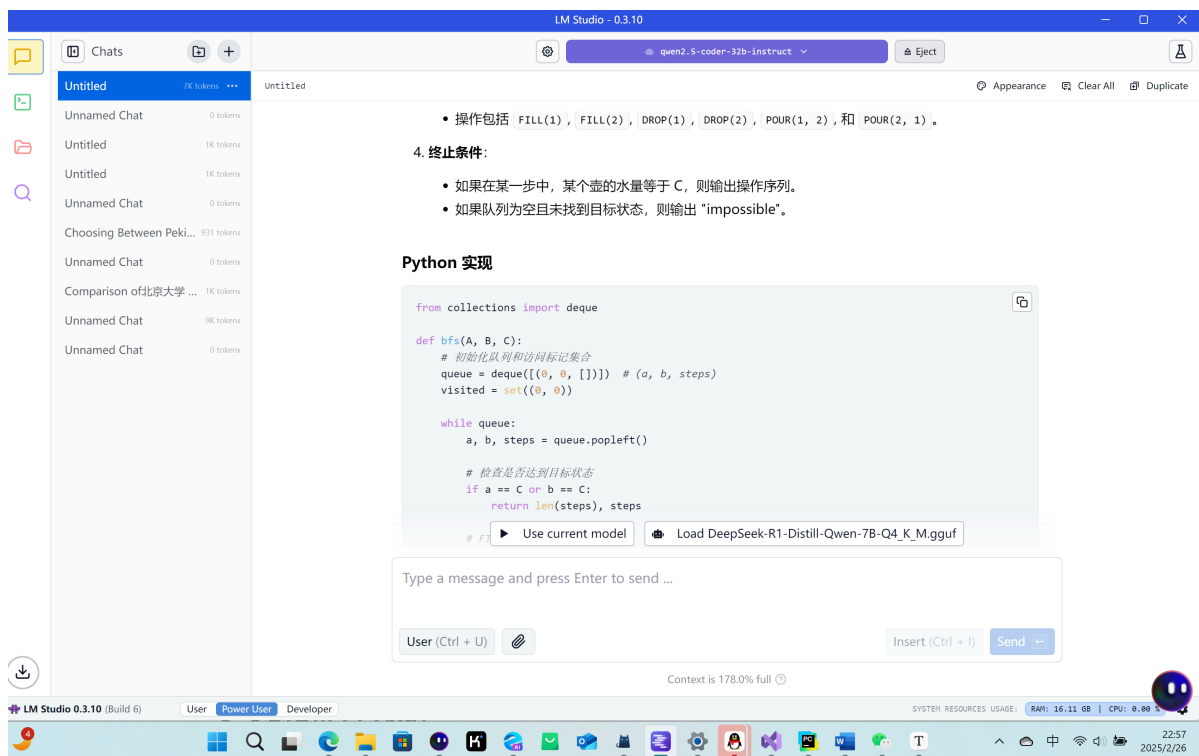
本任务旨在本地环境或通过云虚拟机（如 <https://clab.pku.edu.cn/> 提供的资源）部署大语言模型（LLM）并进行测试。用户界面方面，可以选择使用图形界面工具如 <https://lmstudio.ai> 或命令行界面如 <https://www.ollama.com> 来完成部署工作。

测试内容包括选择若干编程题目，确保这些题目能够在所部署的LLM上得到正确解答，并通过所有相关的测试用例（即状态为Accepted）。选题应来源于在线判题平台，例如 OpenJudge、Codeforces、LeetCode 或洛谷等，同时需注意避免与已找到的AI接受题目重复。已有的AI接受题目列表可参考以下链接：

[https://github.com/GMyhf/2025spring-cs201/blob/main/AI\\_accepted\\_locally.md](https://github.com/GMyhf/2025spring-cs201/blob/main/AI_accepted_locally.md)

请提供你的最新进展情况，包括任何关键步骤的截图以及遇到的问题 and 解决方案。这将有助于全面了解项目的推进状态，并为进一步的工作提供参考。

尝试在本地部署了各种大小的模型，但似乎是因为电脑是轻薄本没有独立显卡的原因，推理型的大模型即使只使用1.5B的也会一直思考而迟迟无法输出答案，较小的模型做题能力较差，基本做不对任何题目。最后部署了Qwen2.5 Coder 32B Instruct感觉能做出一些模板性比较强的题，比如03151 Pots。



## Q6. 阅读《Build a Large Language Model (From Scratch)》第一章

作者：Sebastian Raschka

印象最深的是书中提到GPT这类模型在预训练时的任务仅仅是预测下一个出现的词，但最后却自然涌现出了分类，翻译等复杂的功能。这还是让人感到十分神奇，希望之后能进一步了解这背后更深层次的原理。

## 2. 学习总结和个人收获

---

因为计概的时候就有跟进每日选做，所以这一段时间的编程练习还是比较轻松的。自己又额外参加了力扣的周赛。其中第150场双周周赛的压轴“最短匹配子字符串”要用到KMP算法，比赛时想到了思路但是因为不够熟练没有写完。最近的练习还是侧重于熟练度的提升吧，感觉写多了后速度确实有所上升。