

Assignment #1: 虚拟机, Shell & 大模型

Updated 1317 GMT+8 Feb 20, 2025

2025 spring, Compiled by 颜鼎堃 工学院

作业的各项评分细则及对应的得分情况

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	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
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：

- 类的简单应用

代码：

```

1  from math import gcd
2  class Fraction():
3      def __init__(self, numerator, denominator):
4          self.numerator = numerator
5          self.denominator = denominator
6      def __add__(self, other):
7          n = self.numerator * other.denominator + self.denominator * other.numerator
8          d = self.denominator * other.denominator
9          g = gcd(n, d)
10         return Fraction(n // g, d // g)
11     def __str__(self):
12         return str(self.numerator) + "/" + str(self.denominator)
13
14 a, b, c, d = map(int, input().split())
15 f1 = Fraction(a, b)
16 f2 = Fraction(c, d)
17 print(f1 + f2)

```

代码运行截图 (至少包含有"Accepted")

1760.袋子里最少数目的球

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

思路:

- 最大值的最小值

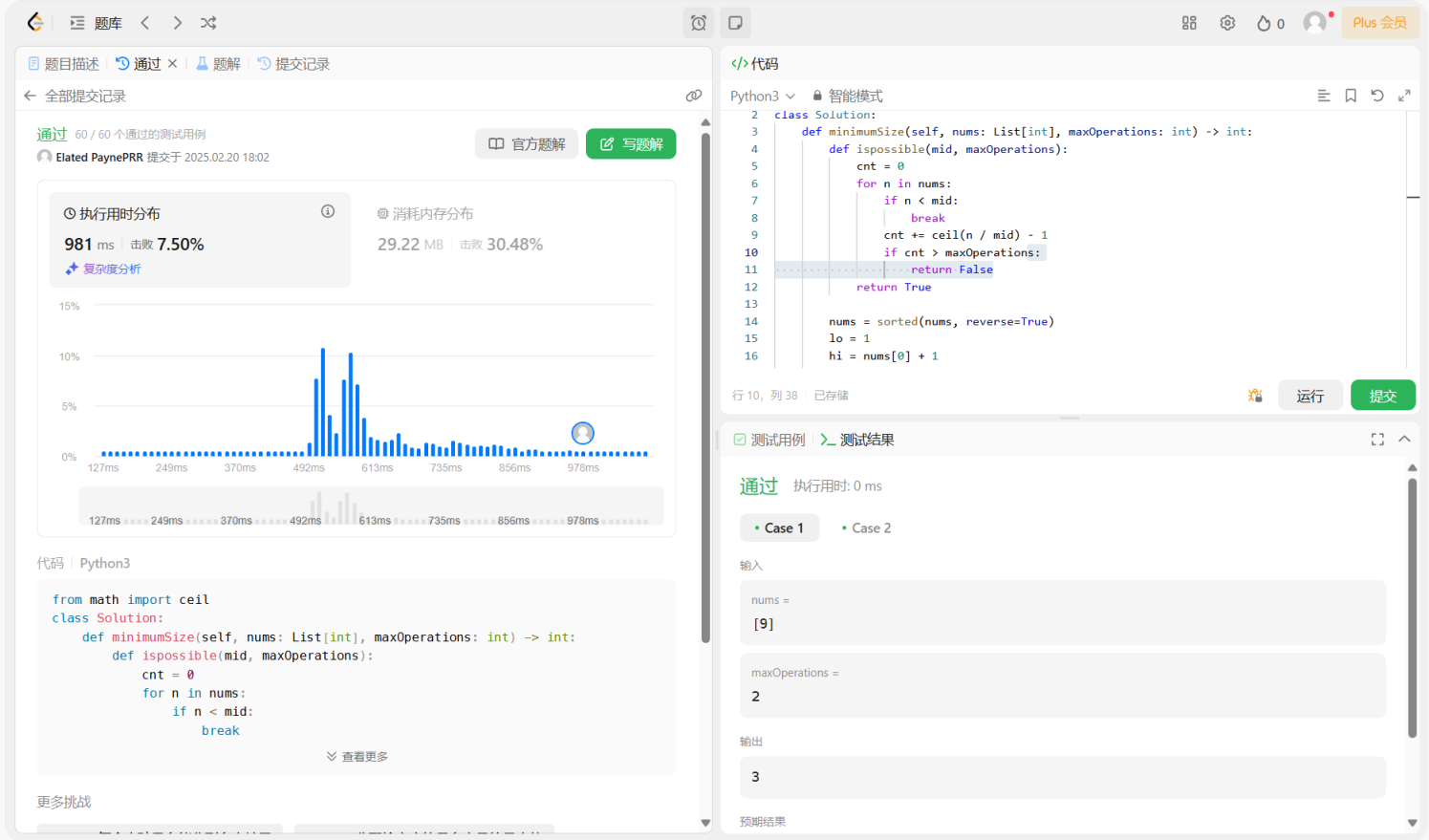
代码:

```

1  from typing import *
2  from math import ceil
3  class Solution:
4      def minimumSize(self, nums: List[int], maxOperations: int) -> int:
5          def ispossible(mid, maxOperations):
6              cnt = 0
7              for n in nums:
8                  if n < mid:
9                      break
10                 cnt += ceil(n / mid) - 1
11             if cnt > maxOperations:
12                 return False
13             return True
14
15         nums = sorted(nums, reverse=True)
16         lo = 1
17         hi = nums[0] + 1
18         while lo < hi:
19             mid = (lo + hi) // 2
20             if ispossible(mid, maxOperations):
21                 hi = mid
22             else:
23                 lo = mid + 1
24         return lo
25
26 if __name__ == '__main__':
27     sol = Solution()
28     print(sol.minimumSize([9], 10))

```

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:

- 和上题类似

代码:

```
1 def ispossible(mid):
2     global N, M
3     cnt = 0
4     m = 0
5     for c in cost:
6         if c > mid:
7             return False
8         m += c
9         if m > mid:
10            cnt += 1
11            m = c
12    return cnt + 1 <= M
13 N, M = map(int, input().split())
14 cost = [int(input()) for i in range(N)]
15 lo = 1
16 hi = sum(cost)
17 while lo < hi:
18     mid = (hi + lo) // 2
19     if ispossible(mid):
20         hi = mid
21     else:
22         lo = mid + 1
23 print(lo)
24
```

Python

OpenJudge

题目ID, 标题, 描述

24n2400011125 信箱 账号

CS101 / 题库 (包括计概、数算题目)

题目 排名 状态 提问

#48316355提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
def ispossible(mid):
    global N, M
    cnt = 0
    m = 0
    for c in cost:
        if c > mid:
            return False
        m += c
        if m > mid:
            cnt += 1
            m = c
    return cnt + 1 <= M
N, M = map(int, input().split())
cost = [int(input()) for i in range(N)]
lo = 1
hi = sum(cost)
while lo < hi:
    mid = (hi + lo) >> 1
    if ispossible(mid):
        hi = mid
    else:
        lo = mid + 1
print(lo)
```

基本信息

#: 48316355

题目: 04135

提交人: 颜鼎堃(24n2400011125)

内存: 7488kB

时间: 375ms

语言: Python3

提交时间: 2025-02-20 20:12:32

©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：

- 麻烦点的排序题

代码：

```
1 from itertools import groupby
2 n = int(input())
3 model = [input().split("-") for i in range(n)]
4 trans = {"B": 1000, "M": 1}
5 for i in range(n):
6     rate = float(model[i][-1][:-1]) * trans[model[i][-1][-1]]
7     model[i].insert(1, rate)
8     model[i] = tuple(model[i])
9 model.sort()
10
11 for key, val in groupby(model, key=lambda t: t[0]):
12     rates = [t[-1] for t in val]
13     print(f"{key}: ", end="")
14     print(*rates, sep=", ")
15
```

Python



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

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和解决方案。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。

- 部署了DeepSeek-R1:1.5b，但智商堪忧
- 遂卸载
- 实际上在Linux云主机上下载Ollama遇到了一些问题，提示网络问题无法下载，但采用如下命令

```
1 curl -O https://ollama.com/install.sh
```

Bash

先将 `install.sh` 下载下来，然后本地运行 `install.sh`，可以正常下载

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

作者：Sebastian Raschka

请整理你的学习笔记。这应该包括但不限于对第一章核心概念的理解、重要术语的解释、你认为特别有趣或具有挑战性的内容，以及任何你可能有的疑问或反思。通过这种方式，不仅能巩固你自己的学习成果，也能帮助他人更好地理解这一部分内容。

- 确实没读

2. 学习总结和个人收获

如果发现作业题目相对简单，有否寻找额外的练习题目，如“数算2025spring每日选做”、LeetCode、Codeforces、洛谷等网站上的题目。

其实上学期已经研究了一下云主机（根本动力是打游戏需要搭建24小时在线的服务器）

但刚开学感觉学习动力不足，心思花少了，后面还需要投入时间