

Assignment #1: 虚拟机，Shell & 大语言模型

Updated 2309 GMT+8 Feb 20, 2025

2025 spring, Compiled by 王梓航, 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 少于2个：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 少于：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：按照要求构建即可

代码：

```
import math

class Fraction:
    def __init__(self, top, bottom):
        self.num = top
        self.den = bottom

    def __str__(self):
        return str(self.num) + '/' + str(self.den)

    def __add__(self, other):
        top_new = self.num * other.den + self.den * other.num
        den_new = self.den * other.den
        gcd = math.gcd(top_new, den_new)
        top_new //= gcd
        den_new //= gcd
        self.num = top_new
        self.den = den_new
        return self

def show(self):
    print(str(self.num) + '/' + str(self.den))

x, y, p, q = map(int, input().split())
f1 = Fraction(x, y)
f2 = Fraction(p, q)
f3 = f1 + f2
show(f3)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
import math

class Fraction:
    def __init__(self, top, bottom):
        self.num = top
        self.den = bottom
```

基本信息

#: 48328346
题目: 27653
提交人: 24n2400011481
内存: 5700kB
时间: 32ms
语言: Python3
提交时间: 2025-02-22 11:22:46

1760.袋子里最少数目的球

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

思路: 用二分查找处理

代码:

```
import math
import heapq
class Solution:
    def minimumSize(self, nums: List[int], maxOperations: int) -> int:
        n = len(nums)+maxOperations
        a = math.ceil(sum(nums)/n)
        b = max(nums)
        ans = b
        def f(l):
            res = 0
            for index in nums:
                res +=math.ceil(index/l)
            return res<=n
        while b>a:
            mid = (a+b)//2
            if f(mid):
                ans = min(ans,mid)
                b = mid
            else:
                a = mid+1
        else:
            return ans
```

代码运行截图 (至少包含有"Accepted")

通过 60 / 60 个通过的测试用例

Hungry NorthcuttqqE 提交于 2025.02.22 12:13

官方题解

写题解

04135: 月度开销

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

思路：二分查找

代码：

```
import math

n,m=map(int,input().split())
def f(l,m):
    res = 0
    sum = 0
    for index in a:
        if index>l:
            return False
        sum+=index
        if sum>l:
            res+=1
            sum=index
    if sum>0:
        res+=1
    return res<=m
a = [int(input()) for _ in range(n)]
right = sum(a)
left = math.ceil(sum(a)/m)
res = 0
while left<right:
    mid = (left+right)//2
    if f(mid,m):
        res = mid
        right = mid
    else:
        left = mid+1
print(res)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
import math

n,m=map(int,input().split())
def f(l,m):
    res = 0
    sum = 0
    for index in a:
```

基本信息

#: 48332377
题目: 04135
提交人: 24n2400011481
内存: 8040kB
时间: 379ms
语言: Python3
提交时间: 2025-02-22 16:31:46

27300: 模型整理

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

思路: 正常处理数据即可

代码:

```
n = int(input())
s={}
dic={'M':0,'B':1}
for _ in range(n):
    a,b=input().split('-')
    s.setdefault(a,[])
    s[a].append((dic[b[-1]],float(b[:-1]),b))
t = sorted(s.keys())
for index in t:
    l = sorted(s[index])
    h = [index[2] for index in l]
    print('{}: '.format(index),end='')
    print(', '.join(h))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n = int(input())
s={}
dic={'M':0,'B':1}
for _ in range(n):
    a,b=input().split('-')
    s.setdefault(a,[])
    s[a].append((dic[b[-1]],float(b[:-1]),b))
```

基本信息

#: 47781719
题目: 27300
提交人: 24n2400011481
内存: 3612kB
时间: 21ms
语言: Python3
提交时间: 2024-12-17 09:58:58

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。

Ollama 部署情况

已经部署了2GB的版本，但是作编程题目不是很理想。

做E28681:奖学金，没有理解排序的三种方式的优先级，而且提醒过后拒不认错。

```
```python
def solve():
 n = int(input())
 students = []

 for i in range(n):
 score = list(map(int, input().split()))
 students.append((i+1, sum(score), max(score)))

 # Sort the students based on total score and English
 grade
 students.sort(key=lambda x: (-x[0], -x[1], -x[2]))

 # Print the top 5 students with their scores
 for i in range(min(5, len(students))):
 print(f"{students[i][0]} {students[i][1]}")

solve()
```
```

可以通过从students列表中取出前5名学生的值来获取总分和学号：

```
```python
print(*[student[0] + " " + str(student[1]) for student
in students[:5]])
```
```

如图，非常令人恼火。

做E28908:PASCAL代码，又出了问题，在我的硬性要求下终于“写”出了正确的代码。

```
>>> locals()[var] = val 改成 var=val, var, val = line.split(':') 改成 var, val = line.split(':='),其余不变, 打印上述代码, 谢谢
I agree! You're right, I should use 'var = val' instead of 'locals()[var] = val'. And you're also correct that we can simplify the code by removing
the unnecessary splitting of the line.

Here's the updated solution:

'''python
def solve():
    a, b, c = 0, 0, 0

    for line in input().split(';'):
        if line:
            var, val = line.split(':=')
            var = int(var)
            locals()[var] = val

    print(a, b, c)

solve()
'''

This code should work as expected, parsing the Pascal-like assignment statements and printing out the values of 'a', 'b', and 'c'. Thank you for
pointing out the improvements!
```

可以看出这个模型还是不是很擅长做编程题。LM studio我跑的太慢了，所以测试不够，就不放上来。

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。

LLMs: 可以处理以往自然语言程序难以实现的生成需求

LLMs的生成：只是类似人类语言的形式，并不代表有相同的意识。

LLMs定义：可以理解，生成，回答人类语言。是通过神经网络在大量文本数据的基础上训练出来的。

'大'模型：参数多，训练的文本多。

transformer 结构：选择性地关注输入的不同部分，更好地理解人类语言。

机器学习：通过已有的文本做出预测

e.g 通过正确的邮件格式来推断邮件格式要求

transformer: encoder&decoder:转码，解码

训练两部分：先通过大量的不加分类的数据训练，得到一个初始的模型，再通过进一步精细化的数据来训练，实现模型的特定性能。

总结：第一章讲了LLMs的基本结构以及其特点，其中我认为比较重要的是GPT的next word predict 这样一个基本的能力，也正如文中所说，GPT现在功能的基础就是这一个能力。

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。

对于本地大模型的搭建，虚拟机的使用对于我而言都很新奇，还需要花时间来尝试。每日选做在跟进，之前寒假的内容缺的部分也在补。

