**《Python程序设计》**

**课程项目作业报告**



**序号及名称：**作业1-校生游戏（学期）

**学号与姓名：**3240103494

**年级与专业：**大一 工商管理

**学 院 (系)：**求是学院

**日期：**2025年4月15日

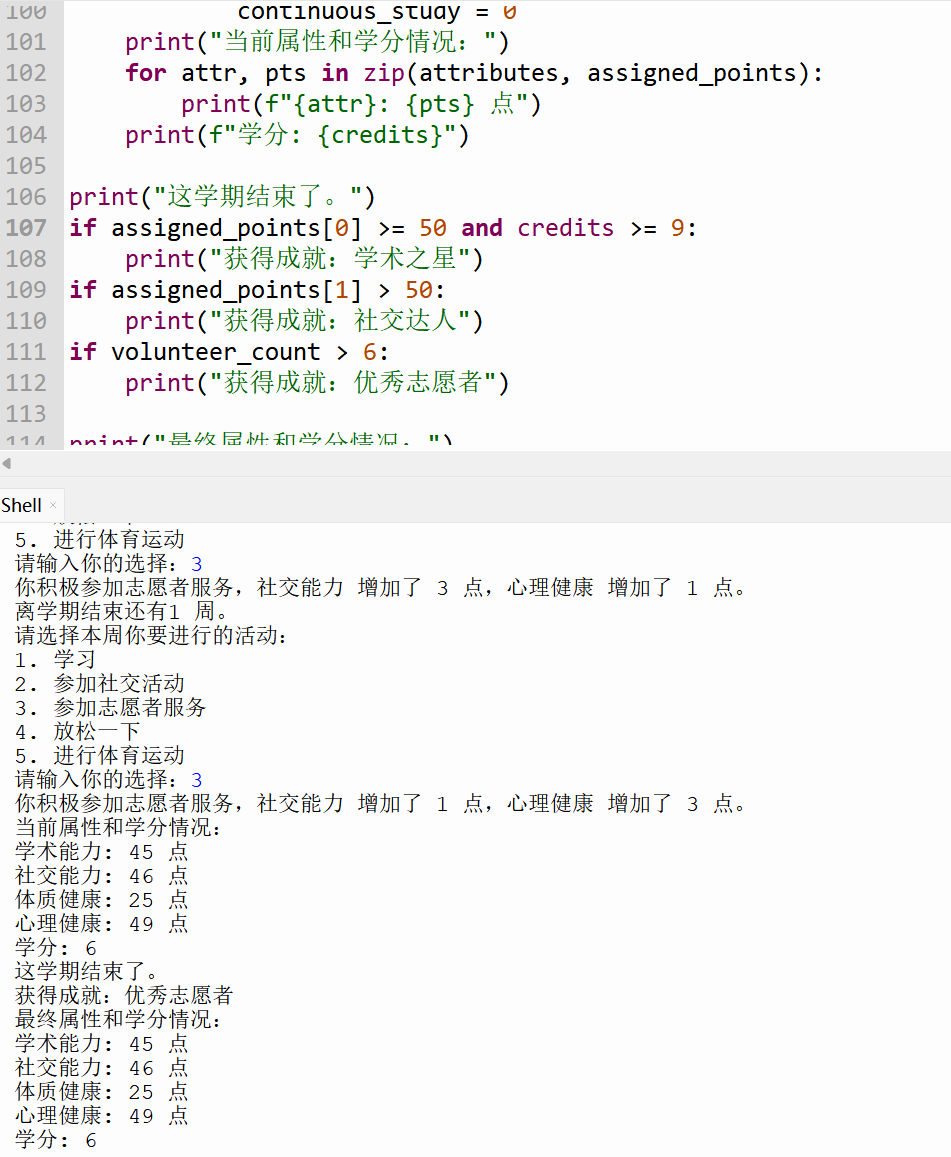
# **作品名称：《校生游戏》**

难度:☆☆☆

## 第一部分 功能实现（20分）

### 代码测试游戏界面（10分）

1.Thonny



1. CodeSculptor3



设计思路游戏逻辑（10分）：

模块化设计：代码通过不同的段落模拟大学生活的不同方面，如学习、社交、休闲、锻炼等，使得游戏逻辑清晰易懂。

玩家互动：通过input()函数获取玩家的选择，增加游戏的互动性。

运气系统：为使游戏更贴合现实，对玩家的属性点进行随机赋予，增加了游戏的不确定性与趣味性。

状态更新机制：游戏通过不断更新玩家的状态来模拟现实生活中的决策后果，并实时更新，便于玩家更好把握自己当前状态从而更好的做下一步决策。

事件关联：选择不同事件所获得的结果看似单一，但是也相互关联，增加了游戏的趣味性，同时也隐晦地提示玩家在大学生活中要做到劳逸结合。

成就系统：对玩家所做的不同选择进行汇总，整体性地给予评价，激发玩家再次游玩的兴趣。

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **自评 分数** | **教师 确认** |
| 游戏界面 （10分） | **10分**：界面完整，各元素清晰，有游戏截图**7-9分**：界面基本完整 ，部分元素需改进**4-6分**：界面简陋但可用**0-3分**：界面不完整或无法使用 | 9 |  |
| 游戏逻辑 （10分） | **10分**：完整实现选择系统和属性影响机制**7-9分**：基本实现核心功能，有小缺陷**4-6分**：部分功能实现，存在明显缺陷**0-3分**：核心功能缺失或无法运行 | 9 |  |

## 第二部分 技术实现（30分）

### 代码质量分析（10）：

此代码通过模拟大学生活中的各种活动（如学习、社交、休闲娱乐等）形成一个简单的校生游戏，展示了大学生生活的一个截面。每项活动都会影响玩家的学术能力、社交能力、体质健康和心理健康，从而影响最终的游戏结果，即学期末的呈现。

1. 变量初始化：游戏开始时，由玩家自主分配学术能力、社交能力、体质健康和心理健康四个属性值。
2. 周循环：使用for循环遍历每一学期中的十六个学习周，每一周中玩家会面临相同的五个活动，但是获得的决策成果却很有可能大相径庭。
3. 志愿服务模块：若玩家选择活动三，即参加志愿服务，各项属性值的提升幅度较其他活动较小，但对学期末的成就判断产生较大影响。
4. 学分增加：玩家获得学分的途径并不单一。通过学习获得学分时，需要选择较多次数的学习；而通过志愿者服务获得学分具有偶然性，其判定的根据是random函数生成的随机数字。
5. 状态更新：根据玩家的活动和选择更新学术能力、社交能力、体质健康和心理健康四个属性值。若体质健康或心理健康的值过低，则会有特殊的影响。
6. 成就判断：在十六次选择全部结束，即十六个学习周结束后的学期末，根据玩家状态进行成就判断，并最后一次完整展示玩家属性值与学分。

### 代码对应的Python知识点和AI使用说明（10分）：

* 变量：使用变量存储和更新玩家的状态信息。
* 控制流：使用if-elif-else多重分支语句和for循环、while循环控制游戏逻辑和流程。
* 输入输出：使用print()和input()函数与玩家交互。
* 随机数生成：使用random.randint()和random.sample()生成属性改变值。
* 列表：建立列表attributes和assigned\_points用于存储属性名称与分配点数。

AI部分：

* enumerate函数：用enumerate()获取列表索引和元素。
* zip函数：用zip()遍历两个列表，以同步获取属性名称与对应点数。

### 技术创新和源码注释及高亮（10分）：

import random# 导入random库，用于生成随机的属性变化值

print("欢迎来到大学校园生活模拟游戏！")# 游戏开始，打印欢迎信息

name = input("请为您的角色起名：")#获取玩家输入的角色名称

#由玩家分配角色的各项状态，包括学术能力、社交能力、体质健康和心理健康四个属性值

#通过预设可能出现的错误输入编写相应程序予以玩家反馈

print("请自由分配你的属性点(请输入整数）")

points = 100

attributes = ["学术能力", "社交能力", "体质健康", "心理健康"]#定义四个属性

assigned\_points = [0] \* 4#初始化四个属性

count = 0#用于记录已分配的属性数量

while True:#while循环，以确保遍历每个属性

for i, attr in enumerate(attributes):

while True:

print(f"当前剩余点数: {points}")

assigned = int(input(f"请输入分配给 {attr} 的点数: "))

if assigned < 0:

print("分配的点数不能为负数，请重新输入。")

continue

if assigned > points:

print(f"分配的点数不能超过剩余的 {points} 点，请重新输入。")

continue

assigned\_points[i] = assigned

points -= assigned#记录剩余可分配的属性点

count += 1

Break

if count == len(attributes):

break#四个属性全部分配完毕时，此段程序结束

print("分配结果如下：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

if assigned\_points[2] < 10 or assigned\_points[3] < 10:

print("你的身体状况很差，被送入了医院。")

print("这学期你在医院里养病。")#体质健康值与心理健康值直接影响了一整个学期的课程

credits = 0#确定学分值，为最后的成就判断提供依据

volunteer\_count = 0#确定志愿服务次数，为最后的成就判断提供依据

hosipitalized = 16#此时十六周都在住院，住院周数确定，为最后的成就判断提供依据

else:#在不住院的情况下开始正常的学期生活

credits = 0

volunteer\_count = 0

hosipitalized = 0#初始化三个成就判断的依据

choices = 16#初始化可供选择的次数

a = 0#为后续的通过学习增加学分的程序设定提供判定依据

continuous\_study = 0#为连续学习影响健康的程序设定提供判定依据

#使用for循环开始玩家的选择历程，以周为单位，遍历每一周

for i in range(choices): print(f"离学期结束还有{choices - i} 周。")

print("请选择本周你要进行的活动：")

print("1. 学习")

print("2. 参加社交活动")

print("3. 参加志愿者服务")

print("4. 放松一下")

print("5. 进行体育运动")

choice\_str = input("请输入你的选择：")

choice = int(choice\_str)#将输入字符串转换为整数，方便后续判定

if choice <= 0 or choice >= 6:

print("请输入有效的选择。")

Continue#输入不合法，让玩家进行重新选择

#各个活动分支根据玩家的选择执行对应逻辑

if choice == 1:

increase = random.randint(1, 5)#随机增加属性值

assigned\_points[0] += increase

print(f"你认真学习，学术能力增加了 {increase} 点。")

a += 1#累计学习次数

continuous\_study += 1

if a % 3 == 0:

print("你完成了一门课，获得三学分。")

credits += 3#每三次学习获得一次学分

if continuous\_study >= 6:#连续学习次数过大，触发随机事件降低体质健康或心理健康

target = random.choice([2, 3])

decrease = random.randint(1, 5)

assigned\_points[target] -= decrease

print(f"你的 {attributes[target]} 降低了 {decrease} 点。")

if assigned\_points[3] < 15:

print("你的心理健康低于十五，你的状态让你很难好好学习")

assigned\_points[0] -= increase#心理健康过低，学习无效，学分和学术能力不增加

elif choice == 2:

increase = random.randint(1, 5)

assigned\_points[1] += increase

print(f"你积极参加社交活动，社交能力增加了 {increase} 点。")

continuous\_study = 0#非学习活动，将连续学习次数清零

elif choice == 3:

volunteer\_count += 1

target1, target2 = random.sample([1, 3], 2)

increase1 = random.randint(1, 3)

increase2 = random.randint(1, 3)

assigned\_points[target1] += increase1

assigned\_points[target2] += increase2

print(f"你积极参加志愿者服务，{attributes[target1]} 增加了 {increase1} 点，{attributes[target2]} 增加了 {increase2} 点。")

continuous\_study = 0

if increase1 == increase2:

credits += 1

elif choice == 4:

increase = random.randint(1, 5)

assigned\_points[3] += increase

print(f"你决定让自己放松一下。")

print("你度过了比较轻松的一周，心理健康增加了 {increase} 点。")

continuous\_study = 0

else:

increase = random.randint(1, 5)

assigned\_points[2] += increase

print(f"你的体质健康值增加了 {increase} 点。")

continuous\_study = 0

if assigned\_points[2] < 15:

print("你的体质健康已经低于15，不幸生病，本周只好在医院度过。")

choices -= 1

assigned\_points[2] = 16

continuous\_study = 0

print("当前属性和学分情况：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

print(f"学分: {credits}")

print("这学期结束了。")

if assigned\_points[0] >= 50 and credits >= 9:

print("获得成就：学术之星")

if assigned\_points[1] >= 50:

print("获得成就：社交达人")

if volunteer\_count >= 4:

print("获得成就：优秀志愿者")

if hospitalized >= 3:

print("获得成就：脆皮大学生（请注意身体哦）")

print("最终属性和学分情况：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

print(f"学分: {credits}")

elif choice == 3:

volunteer\_count += 1

target1, target2 = random.sample([1, 3], 2)

increase1 = random.randint(1, 3)

increase2 = random.randint(1, 3)

assigned\_points[target1] += increase1

assigned\_points[target2] += increase2

print(f"你积极参加志愿者服务，{attributes[target1]} 增加了 {increase1} 点，{attributes[target2]} 增加了 {increase2} 点。")

continuous\_study = 0

if increase1 == increase2:#根据random函数生成的随机数判定玩家是否能通过运气增加学分

credits += 1

elif choice == 4:

increase = random.randint(1, 5)

assigned\_points[3] += increase

print(f"你决定让自己放松一下。")

print("你度过了比较轻松的一周，心理健康增加了 {increase} 点。")

continuous\_study = 0

else:#没有其他更多的选项，直接使用else而不使用elif语句

increase = random.randint(1, 5)

assigned\_points[2] += increase

print(f"你的体质健康值增加了 {increase} 点。")

continuous\_study = 0

if assigned\_points[2] < 15:

print("你的体质健康已经低于15，不幸生病，本周只好在医院度过。")

choices -= 1

assigned\_points[2] = 16

continuous\_study = 0#体质健康低于十五时触发生病住院事件，并减少一次选择次数

#每周选择结束后实时更新玩家状态值

print("当前属性和学分情况：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

print(f"学分: {credits}")

#选择次数清零时，进行学期总结

print("这学期结束了。")

#根据最终属性，学分，志愿服务次数及住院次数进行成就判断并发放成就

if assigned\_points[0] >= 50 and credits >= 9:

print("获得成就：学术之星")

if assigned\_points[1] >= 50:

print("获得成就：社交达人")

elif choice == 3:

volunteer\_count += 1

target1, target2 = random.sample([1, 3], 2)

increase1 = random.randint(1, 3)

increase2 = random.randint(1, 3)

assigned\_points[target1] += increase1

assigned\_points[target2] += increase2

print(f"你积极参加志愿者服务，{attributes[target1]} 增加了 {increase1} 点，{attributes[target2]} 增加了 {increase2} 点。")

continuous\_study = 0

if increase1 == increase2:

credits += 1

elif choice == 4:

increase = random.randint(1, 5)

assigned\_points[3] += increase

print(f"你决定让自己放松一下。")

print("你度过了比较轻松的一周，心理健康增加了 {increase} 点。")

continuous\_study = 0

else:

increase = random.randint(1, 5)

assigned\_points[2] += increase

print(f"你的体质健康值增加了 {increase} 点。")

continuous\_study = 0

if assigned\_points[2] < 15:

print("你的体质健康已经低于15，不幸生病，本周只好在医院度过。")

choices -= 1

assigned\_points[2] = 16

continuous\_study = 0

print("当前属性和学分情况：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

print(f"学分: {credits}")

print("这学期结束了。")

if assigned\_points[0] >= 50 and credits >= 9:

print("获得成就：学术之星")

if assigned\_points[1] >= 50:

print("获得成就：社交达人")

if volunteer\_count >= 4:

print("获得成就：优秀志愿者")

if hospitalized >= 3:

print("获得成就：脆皮大学生（请注意身体哦）")

print("最终属性和学分情况：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

print(f"学分: {credits}")

if volunteer\_count >= 4:

print("获得成就：优秀志愿者")

if hospitalized >= 3:

print("获得成就：脆皮大学生（请注意身体哦）")

#打印最终属性和学分

print("最终属性和学分情况：")

for attr, pts in zip(attributes, assigned\_points):

print(f"{attr}: {pts} 点")

print(f"学分: {credits}")

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **自评 分数** | **教师 确认** |
| 代码质量 （10分） | **10分**：代码结构清晰，命名规范，有注释**7-9分**：代码基本规范，结构较清晰**4-6分**：代码可读性一般，结构需改进**0-3分**：代码混乱，难以理解 | 9 |  |
| AI辅助使用 （10分） | **10分**：AI使用率<30% ，并有明确说明使用方式**7-9分**：AI使用率<30%，  说明不够详细**4-6分**：AI使用率接近30%，缺乏说明**0-3分**：AI使用率 >30%或无说明 | 10 |  |
| 技术创新 （10分） | **10分**：使用了高级技术或创新解决方案**7-9分**：有一定技术亮点**4-6**  **分**：使用基本技术实现功能**0-3分**：技术实现简单，缺乏亮点 | 8 |  |

## 第三部分：学习成果（20分）

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **自评 分数** | **教师 确认** |
| 能力收获（ 10 分） | **10分**：详细描述3项以上具体能力提升**7-9分**：描述2-3项能力提升 **4-6分**：简单描述1-2项能力提升**0-3分**：未描述或描述不具体 | 10 |  |
| 1. 编程能力：   从课上抄写老师给出的代码却依然运行报错，到自己独立编写一个较为完整的从=程序，我的编程能力毫无置疑是得到了很大提升。在本次编写中，我不仅是在独立的反复试错里修正完善if-elif-else的多重分支和for循环的编写，对行前缩进有了更加深刻的理解，还在AI的帮助下更深层次地了解了列表，更完整的认识了enumerate()函数和zip()函数。   1. 逻辑设计能力：   我对编程的理解从个体一点点走向了组合，能够更加熟练地使用分支与循环，对复杂逻辑的分层实现更加有把握有信心，对嵌套的理解也更加深刻，最终呈现为上述程序。各种状态程序在不同分支中都有出现，要如何安排其初始值设定的合适位置，也是我反复思考并最终确定的。   1. 解决问题的信心和勇气：   在此次作业中，我对我的编程能力更有信心，更相信自己可以完成python的相关学习。不只是信心，我更愿意去挑战的这份勇气是我觉得我在这次作业里最大的提升。   1. 相关编程思维的有效提升：   不只是前文有所涉及的程序化逻辑化思维，还有在编写程序时让我更加深刻地认识到的交互性思维。我的程序从非常简陋的框架走到如今的呈现结果，是我反复试玩，从玩家角度去思考如何增加游戏可玩性趣味性的结果。但同时，作为程序设定者还希望有有所出人意料的地方，所以增加了住院的相关设计。  另外，还有模块化思维的提升。本次程序设定先是属性分配模块，再是判定后分裂为住院模块与不住院模块，在不住院模块里又分为五个活动模块。这种思维是的代码更加清晰易懂，也更逻辑化系统化。 | | |  |
| 时间投入与产 出（ 10分） | **10分**：详细记录时间投入，产出比高**7-9分**：有时间记录，产出比合理 **4-6分**：时间记录不详细，产出一般**0-3分**：无时间记录或产出比 低 | 9 |  |
| 第一次机房课，利用AI生成极长的校生游戏代码，为本次作业奠定了基本的模块框架，即五个活动的选择。  4.8，在得知此次作业后，在AI生成的代码基础上思考自己的游戏代码编写，创新了属性分配环节。  4.9，初次完成五个活动的代码编写。  4.11，增加心理健康属性，产生住院相关设计的想法，并第一次尝试。  4.14，将两个住院不住院模块用if-else加以联系，解决了第一次代码住院后依旧能进行学习周选择的问题。  4.15，借助AI修正代码中首行缩进的问题，完善成就判断模块。 | | |  |

## 第五部分：教学反馈（ 10分）

|  |  |  |  |
| --- | --- | --- | --- |
| **分项** | **评分标准** | **自评 分数** | **教师 确认** |
| 反馈质量 （10分） | **10分**：提供具体、有建设性的反馈 ，至少3条**7-9分**：提供较有价值的反 馈 ，2-3条**4-6分**：提供一般性反馈**0-3分**：反馈模糊或缺失 | 8 |  |
| 1. 希望老师能更加清楚地帮助相关函数的理解，有时候照抄了代码能够加以运行但无法理解其运行逻辑。 2. 希望能够给出更多的函数，因为在AI辅助学习的过程中发现了很多自己并不了解的函数，比如try函数。 3. 教学时引入的很多相关网站很有吸引力也很有趣，但是很多时候并不会操作，在基础功能的实现上会觉得同质化严重，有时甚至来不及打开或打不开。 | | |  |