

Tutorial 1. 多大CSC assignment 写作指南 1.0

Bluekey Education

如何分析逻辑，并将逻辑转换为code，再将code转化为分数

1. 搞清楚这个assignment & project大概是要做成一个什么东西

首先，需要对a/p做一个quick review，了解这个a/p：

- (1) 大概是要完成一个什么样的东西
- (2) 大致蕴含了哪些知识点
- (3) 理解其基本逻辑

举例

如果要完成一个puzzle游戏，那么这个程序的玩法就很需要理解

(1) 玩法是怎样

- a. 棋盘config
- b. player人数
- c. 每个player每一步的可选方案
- d. 输赢判定条件

(2) 这些玩法背后其实隐藏了什么知识点？

- a. string parsing
- b. 2D list / for loop / loop
- c. input/output
- d. if / else

(3) 基本逻辑是什么？

- a. 每名player按照left words去猜，猜对了得分猜错了不得分
- b. 电脑根据player所输入的结果在board上分析，一模一样才得分
- c. 所有词猜测完毕则结束，得分高的player获胜

2. 分析具体要实现哪些功能

在大致了解这个a/p的内容后，需要进行需求分析：

- (1) 即搞清楚这个a/p大概要实现哪些功能
- (2) 对实现这些功能的方法有没有特地的要求

举例

还是以以一个puzzle游戏为例。

(1) 大概要实现的功能

AI side:

- a. board的initial config (初始化配置)：每次player猜测前都需要把这样一个由字符组成的board打印出来
- b. 针对每次player输入的parsing和判定

- c. 每一次操作point的计算
- d. 对words list的更新及游戏结束判定
- e. update player的分数
- f. 游戏赢家的判定

UI side:

- a. 每一位player的输入
- b. 不同player的toggle (切换)

error handle:

player的invalid(非法)输入:

- a. 输入word不在words list中
- b. 输入的row number越界
- c. 输入的word在board中search越界

(2) 特殊方法的要求

通过读descriptions得知, 有些方法有实现的要求, 比如:

- a. reverse不能使用python build-in
- b. 需要使用helper function, 不能重复定义一些功能

Assignment Project Exam Help

3. 找到这些功能的关系, 根据相关性进行分类

这一步很关键, 需要对功能分析给出其优先级, 再根据优先级进行实现。

高优先级的功能具有以下特点:

- (1) 是其他功能的基础, 或被多次复用
- (2) 实现难度不高

举例

依然以puzzle为例。

高优先级的任务:

AI side:

board的init和print: 必须先实现, 不然无法调试程序

word search: 根据player的输入判定是否为正确的pattern, 核心功能

UI side:

player的输入: 核心功能, 可以被两个player复用

error handle:

必须首先实现error handling的程序, 不然当开发进行到后期代码难度增大以后这部分就很难设计好了

4. 对分析好的功能进行一步一步实现

理解功能需要后, 就需要对功能进行实现了。实现的过程分为代码设计和算法设计两部分

- (1) 代码设计: 分析需要哪些函数, 哪些class, 哪些data structure, 以及如何组织这些东西来实现复杂的功能。代码设计的目的是让程序:
 - a. 通过正确的设计减少可能的错误
 - b. 增加代码的可读性和可维护性

c. 提高编写代码的效率

(2) 算法设计：如何设计一个算法，使得功能在实现的同时尽量时间短，同时占用的资源少

Note：代码设计和算法设计是紧密相连的，你中有我，我中有你。好的代码设计可以实现更好的算法设计，而好的算法设计也需要符合当前的代码设计。因此需要在写代码前先在脑海中，或者在纸上、电脑上进行设计，避免盲目低效地写码。

举例

还是以puzzle为例

(1) 代码设计

a. 需要的variable和constant

需要的const都列在了assignment中，主要用于：

1. 控制方向
2. 控制加分
3. 控制player

b. 需要的基本函数

通过观看函数的descriptions可以知道需要的函数有：

get_current_player：返回当前player的ID

get_winner：通过比较P1, P2的分数来判定输赢

reverse：反转一个string，并且不能用python的built-in

get_row：返回一个puzzle string中某一行的sub_string

get_factor：通过方向来返回相应的用于计分的factor

get_points：通过方向和剩余word数来对当前的player计分。这个函数显然需要call get_factor function作为helper function

check_guess：根据player的输入做出相应的判断。显然这需要get_points这个函数作为helper function，同时要判断给定的pattern是否在string-base里。

(2) 代码实现 (demo)

以下将展示一些demo来实现上述的function

```
# reverse string
```

```
# 由于不能使用python自带的function和methon，因此以下的方法是不行的
```

```
def reverse(s):
```

```
    return s[::-1]
```

```
# 必须要自己设计一个reverse方法，利用loop，比如
```

```
def reverse(s):
```

```
    return ''.join([s[i] for i in range(len(s)-1, -1, -1)])
```

```
# 有了reverse，就可以通过只定义forward和up的方式来实现backward和down
```

```
# 的功能。大大增加了效率！
```

(3) 算法分析

算法分析的目的，就是为了尽量以最小的时间，空间资源来完成特定的任务。

在这次assignment中，由于问题比较直接，算法分析能做的提升空间不大。但是，依然有一些可以值得思考的点：

- a. 在判定player是否能得分时（player输入与board上内容match），当player输入错误，能否避免程序做不必要的判定？
- b. 是否可以不把全部board load如内存进行操作？

5. 撰写unit-test测试代码功能

这是作为一个programmer的基本要求，即对自己所做的code进行基本的测试，从而确保基本功能是准确无误的。unit-test需要尽量cover所有可能的情况，并针对这些情况设想正确的输出形式。

举例

依然以puzzle为例

针对section 4中的implementation，需要适当的写一些unit-test来测试功能的正确性。同时test要尽量涵盖所有可能的情况。

以下代码（demo）展示了部分unit-test的写法：

```
# 测试一下自己的reverse() function是否正确
s1 = 'abc'
s1_ref = 'cba'
s2 = '' # corner case
s2_ref = ''
assert(s1_ref == reverse(s1))
assert(s2_ref == reverse(s2))
```

由于篇幅原因，很多demo无法在此展示。对于这部分，欢迎有兴趣的同学课后交流讨论。

Note: 得分关键

- 1、函数的写法必须有五要素：

Example, Header, Description, Body, Function test

写整洁规范的代码，对于拿到高分非常有帮助！

- 2、一定要考虑到corner-case,并且使自己的unit-test尽量测试这些可能的情况

比如最常见的，当用户的输入不合法时，必须有相应的应对机制。如果缺乏这样的机制，程序就会崩溃，可能导致严重后果。

3、多些comment，有助于阅卷TA理解（但是也不要写废话，尽量在一些关键步骤写comment，如if，helper function等）

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder