

报告

组员：姜锐漳，郭宸淏，黄俊维

程序功能

- 1. 通过文件对课程信息进行管理，有存档功能。
- 2. 预选课可以在时间上有重复（与选课网不同），可以根据兴趣等推荐选课，以及投点方式，结果保存在 `points.txt` 中。

项目各模块与类设计细节

通过 PySide6 实现。

设计了三个窗口，`mainwindow`，`addcourse` 和 `modifycourse`。

`mainwindow` 显示 课程列表 以及 预选结果，还有可以点击不同按钮实现 添加/修改课程信息 和 推荐选课 的功能。

`addcourse` 在左侧输入课程信息，在右侧选中上课时间，实现添加课程。

`modifycourse` 在选中一门课后，用与 `addcourse` 中完全一致的方法修改课程信息，也可以删除课程。

UI 设计方面，初始界面是我们整个项目的封面，开始选课后各界面在做的尽可能横平竖直地整齐之余，还通过style.qss文件设置字体、颜色、间距等格式来美化各个控件进而美化整个界面。

核心的类有 `Course` 和 `Res`，分别为一门课的信息，和一组选课方案的信息。以及选课的限制用 `Req` 类来表示

`Req` 的成员有

成员名称	类型	记录信息
<code>ap</code>	<code>int</code>	可投总点数
<code>lb</code>	<code>int</code>	选课学分下限
<code>ub</code>	<code>int</code>	选课学分上限

`Course` 的成员有

成员名称	类型	记录信息
<code>assignpoint</code>	<code>int</code>	推荐投点，用户不可修改
<code>chosen</code>	<code>int</code>	已选人数
<code>course</code>	<code>str</code>	课程名称
<code>index</code>	<code>str</code>	课程编号 256-bit 转为 hex
<code>intension</code>	<code>int</code>	上课意愿值，用户填写，0 ~ 99

成员名称	类型	记录信息
limit	int	限选人数
must	bool	是否为必选
point	int	课程学分
schedule	list[(int, int)]	上课时间, (day : 1 ~ 7, time : 1 ~ 12)
selected	bool	是否选课
teacher	str	教师名称

Res 的成员有

成员名称	类型	记录信息
courses	list[Course]	一组选课方案
val	float	本方案权值

投点分配方法：

1. 计算每门非必选课的权重，如果 $\text{chosen} < \text{limit}$ 那么权重为 0, 否则为 $\frac{\text{chosen} - \text{limit} + 1}{\text{limit}}$
2. 根据权重分配可投总点数 ap ，如果有多余的点数，那么随机分配给若干门课 1 点

权值的计算方法如下：

1. 用户的选课意愿：对于非必选课，奖励 $6 \times \text{intension} \times \text{point}$
2. 早八课惩罚：假设早八课数量为 s ，那么惩罚 $80 \times s^2$
3. 连堂课惩罚：假设连堂课数量为 t ，那么惩罚 $50 \times t$
4. 每日空闲奖励/繁忙惩罚：对于周一到周日的每一天，假设上课数量为 c ，若 $c = 0$ 那么奖励 1000，若 $c \leq 2$ 那么奖励 500，若 $c > 6$ 那么惩罚 $15 \times (c - 6)^3$
5. 选课概率系数：计算每门课能选上的概率并全部乘起来得到 p ，前四步计算的权值再乘上 \sqrt{p} 。一门课能选上的概率为 $\frac{\text{assignpoint} \times \text{limit}}{\text{assignpoint} + 10 \times \text{chosen}}$ ，这里假设其他同学投点为 10

小组成员分工情况

由于开始时没意识到 python 写 Qt 的的便捷性，所以时先写了不少 c++ 的代码，然后又改成了 python。

郭宸溥负责 Course, Res, Req 的设计和实现，以及实现了 c++ 版本的 organize（推荐选课功能）。

姜锐漳负责实现最初的 ui 设计，槽和信号等的处理，以及将郭宸溥实现的所有功能重构成 python 代码。

黄俊维负责实现 ui 的设计和美化，以及项目 bug 和细节的检查。

以及因为网络问题，一开始并不是所有人都使用 github。

项目总结与反思

实现项目的分工还算是合理，没有人在完成项目时摆烂。

我们的项目并不是什么很惊艳的项目，类似的项目也有很多小组在做，我们这个只能说是 Qt 的简单运用。

对于课程信息，其实可以用 SQL 等工具进行更加高效的管理，这是一个可以改进的地方。

还有就是课程信息导入这一块，还可以用 pyppeteer 和 bs4 等工具进一步贴近真实使用场景。

总的来说本次大作业让我们对 Qt 有了许多了解，掌握了用 Qt 实现一个客户端的基本技能，以及用 github 来管理文件的能力。以后如果想要自己写一个 app 也有了入手的切口，还是很有收获的。