时间窗口算法:

来者把队首踢掉,队列第二个变成新队首,新队首的时间由来者的进入时间决定:100-t3-t2.

计时器正常结束,新队首的时间: t2-t1

队列为空: 100分钟

进入队列的算法:

空:进入队列

非空: 与队列中的班是否存在交集?

存在 节次是否与有交集的一样

一样:不能进

不一样:踢掉不一样的,然后进去

没有 直接进入

节次号的划定:

[sectime2]

# 第一节课的开始和结束时间

sec1 = 8:00-10:05

# 第二节课的开始和结束时间

sec2 = 10:05-12:00

# 第三节课的开始和结束时间

sec3 = 14:00-16:05

# 第四节课的开始和结束时间

sec4 = 16:05-18:00

# 第五节课的开始和结束时间

sec5 = 19:00-21:05

#剩余的为0节次

自助考勤的算法:

老师开启考勤,生成空的文件.

学生参与考勤,选择自助考勤,如果最新的记录为True,不接受他的考勤信息,不是True得到 “你是你,你在这”的判定结果,然后生成:checkinType:Auto. checkinResult:None,IsSucc:True/False.

教师开启抽点考勤,生成随机考勤的学生微信号列表,由学生微信号列表,生成学生记录列表:CheckinType:Random,checkinResult:None,isSucc:None,将学生记录列表追加到文件中

学生参与考勤,选择随机考勤,如果最新的记录为Tru,,不接受他的考勤信息,不是True得到 “你是你,你在这”的判定结果,然后生成:checkinType:Random. checkinResult:None,IsSucc:True/False.

考勤结束,执行getResult

for students ,如果detail文件中不存在, 学号相同且为Auto

1则记录为缺勤

2 存在: 是否为false 如果为false 记录为缺勤

为true 调用是否迟到函数,否 出勤, 否 迟到

将detail中的Random 找出来 记录为 缺勤列表.

for students,在缺勤列表, 最后一条记录为False或None

找到自助考勤的结果,如果之前的是出勤,,改成早退,,之前的是迟到,改为迟到早退,,,之前的是缺勤,..不变,,如果最后一条记录是True ,不变化.

完成后, 小更新一次.只更新最新的次序号,

先生成考勤结果列表,从detail文件中读取,最后一次的,考勤结果,与学号,写入文件中,最终生成一个列表,然后将列表传入小更新函数

小更新([{考勤结果,考勤学生学号}],次序号),遍历文件,如果学号和列表中的某一个相同,更新.捕获异常,文件遍历完却在列表中找不到该学生,和 学号中存在 但是文件中不存在的异常

请假批准的算法:

因为有假条提交的证据,所以,考勤结束的时候需要教师手工确认,确认完成后开始进行扫尾工作,确认的时候生成请假学生列表,然后,老师依次处理,处理完成之后,追加到文件中,考勤结果为请假.

教师修改某条记录:

生成考勤对象,但是不生成文件,而是调用修改方法.

修改后调用小更新方法

大更新算法: