

Homework X

外卖配送调度优化

Data Structures and Algorithms — Lecture 07 配套作业

1 问题描述

某外卖平台在一个矩形区域 $[0, L] \times [0, W]$ (单位: 公里) 内提供配送服务。平台拥有 n 名快递员, 所有快递员初始位于坐标原点 $(0, 0)$, 在营业时间开始时 (第 480 分钟) 上线。

快递员匀速移动, 速度 $v = 0.5$ km/min; 距离使用曼哈顿距离:

$$d = |\Delta x| + |\Delta y|. \quad (1)$$

每笔订单包含取货点 (s_x, s_y) 和送达点 (e_x, e_y) 。订单从下单时刻起 **30 分钟内** 必须送达 (到达送达点), 超时则该订单失败、不计收入。每成功送达一单收入 **10 元**。

1.1 订单类型

类型	说明
预订单 (pre-order)	前一天生成, 共 m 笔。算法在营业前即可知晓全部预订单信息, 可据此预先规划路线。
即时订单 (instant order)	营业时间内实时到达, 算法只有在该订单到达时刻才获知其信息。

1.2 快递员行为模型

1. **一次只配送一单**: 快递员必须完成当前订单 (送达或放弃) 后才能接下一单。
2. **配送流程**: 接单 \rightarrow 前往取货点 \rightarrow 取货 \rightarrow 前往送达点 \rightarrow 送达。
3. **闲置规则**: 未被分配订单的快递员停留在最近一次送达位置 (初始为原点), 不自主移动。
4. **预订单提前出发**:
 - 快递员可在自身空闲时刻即向预订单取货点出发 (即使该时刻早于订单时间)。
 - 若快递员在订单时间之前到达取货点, 需等到订单时间方可取货。
 - 计算公式:

$$\text{delivery_time} = \max(\text{arrive_at_pickup}, \text{order_time}) + \frac{\text{dist}(\text{pickup}, \text{delivery})}{\text{speed}}.$$

5. 即时订单:

- 快递员只有在订单到达后才得知信息，因此最早出发时刻为

$$\max(\text{available_time}, \text{order_time}).$$

- 计算公式:

$$\begin{aligned} \text{delivery_time} = & \max(\text{available_time}, \text{order_time}) + \frac{\text{dist}(\text{courier_pos}, \text{pickup})}{\text{speed}} \\ & + \frac{\text{dist}(\text{pickup}, \text{delivery})}{\text{speed}}. \end{aligned}$$

2 编程要求

- 程序从标准输入 (stdin) 读取数据，向标准输出 (stdout) 写入结果。
- 支持的编程语言: **Python (.py)**、**C++ (.cpp)**、**C (.c)**、**Java (.java)**。
- 时间限制: 每个测试点 **30 秒**。
- 不得使用网络、文件 I/O 等外部资源，仅通过 stdin/stdout 交互。

3 输入格式

```
L W n m
id_1 t_1 sx_1 sy_1 ex_1 ey_1
id_2 t_2 sx_2 sy_2 ex_2 ey_2
...
```

- 第一行:** $L W n m$
 - L, W : 区域长、宽 (浮点数)
 - n : 快递员数量 (正整数)
 - m : 预订单数量 (非负整数)
- 随后若干行:** 每行一笔订单
 - 前 m 行为预订单，之后为即时订单，均按 t (下单时间) **非递减**排列
 - id**: 订单编号 (从 1 开始的正整数，按出现顺序递增)
 - t**: 下单时间 (浮点数，分钟，范围 [480, 1200])
 - sx sy**: 取货点坐标; **ex ey**: 送达点坐标
 - 所有坐标满足 $0 \leq x \leq L, 0 \leq y \leq W$

4 输出格式

```
id_1 courier_id_1 delivery_time_1 success_1
id_2 courier_id_2 delivery_time_2 success_2
...
total_completed total_revenue
```

- 每笔订单一行，顺序与输入一致：
 - id: 订单编号
 - courier_id: 分配的快递员编号 ($1 \sim n$)，若不分配则为 0
 - delivery_time: 送达时间 (浮点数，保留 2 位小数)；不分配时为 -1.00
 - success: 1 表示成功 ($\text{delivery_time} \leq \text{order_time} + 30$)，0 表示失败
- 最后一行: total_completed total_revenue
 - total_completed: 成功送达的总单数
 - total_revenue: 总收入 ($= \text{total_completed} \times 10$)

5 约束

参数	范围
L, W	$1 \leq L, W \leq 100$
n (快递员数)	$1 \leq n \leq 100$
m (预订单数)	$0 \leq m \leq 1000$
总订单数	$1 \leq \text{total} \leq 5000$
速度 v	0.5 km/min (固定)
超时阈值	30 min (固定)
每单收入	10 元 (固定)
营业时间	480 ~ 1200 min (8:00~20:00)
坐标精度	小数点后 2 位
$\text{dist}(\text{pickup}, \text{delivery})$	≤ 15.00 km (保证单独配送可达)

6 评判规则

1. 格式校验: 输出行数、字段数量必须正确。
2. 物理可行性: 评判程序根据快递员状态独立重算每笔订单的最早送达时间。学生报告的 `delivery_time` 与重算值的误差不得超过 ± 0.1 min。
3. 逻辑一致性:

- `courier_id` 必须在 $[0, n]$ 范围内。
- 同一快递员不可同时配送两笔订单（时间线不冲突）。
- `success` 字段须与 $\text{delivery_time} \leq \text{order_time} + 30$ 一致。
- `total_completed` 和 `total_revenue` 须与各行统计一致。

4. 评分:

- 验证不通过 \Rightarrow 该测试点 0 分。
- 验证通过 \Rightarrow 得分 = $\frac{\text{student_completed}}{\text{reference_completed}} \times 100$ （上限 120 分）。

7 示例

7.1 输入

```
10.00 10.00 3 2
1 480.00 2.00 3.00 5.00 6.00
2 500.00 8.00 7.00 8.00 9.00
3 510.00 1.00 1.00 4.00 5.00
4 530.00 7.00 2.00 9.00 8.00
```

7.2 输出（参考解结果，非唯一正确答案）

```
1 1 502.00 1
2 1 514.00 1
3 2 528.00 1
4 2 558.00 1
4 40.00
```

7.3 验算说明

（快递员均从 $(0,0)$ 出发、时刻 480 上线，曼哈顿距离。）

订单	快递员	路径	关键时间	截止	结果
1 (预订单, $t=480$)	C1	$(0,0) \rightarrow (2,3) \rightarrow (5,6)$	$\max(480+10, 480)+12$ $= 490+12 = 502.00$	510	✓
2 (预订单, $t=500$)	C1	$(5,6) \rightarrow (8,7) \rightarrow (8,9)$	$\max(502+8, 500)+4$ $= 510+4 = 514.00$	530	✓
3 (即时, $t=510$)	C2	$(0,0) \rightarrow (1,1) \rightarrow (4,5)$	$\max(480, 510)+4+14$ $= 510+18 = 528.00$	540	✓
4 (即时, $t=530$)	C2	$(4,5) \rightarrow (7,2) \rightarrow (9,8)$	$\max(528, 530)+12+16$ $= 530+28 = 558.00$	560	✓

注：不同算法策略可产生不同的合法输出。评判系统验证物理可行性与逻辑一致性。

8 提交要求

1. 代码文件：实现调度算法的可执行源代码（从 stdin 读取，stdout 输出）。
2. 书面报告应包含：
 - 算法设计思路说明 (Description of the implementation approach)
 - 性能测试结果与分析 (Performance test results and analysis)
 - 遇到的困难与解决方案 (Difficulties encountered and solutions)
3. 命名规则：HWX-ClassXX-学号-姓名，例如 HWX-Class01-12345678-张三
4. 提交邮箱：chengyuma@stu.xjtu.edu.cn
5. 截止时间：2026 年 5 月 5 日 24:00