

项目名称：菜鸟驿站送货小车路径规划

项目背景

菜鸟驿站的送货小车总是在校园的路上忙忙碌碌，但是它有时候看起来不太聪明的样子。本项目想让它变得聪明一点：把排队等待配送的包裹装车，规划路线，完成送货。

项目的主要功能

本项目要实现的主要功能如下

1. 维护驿站快递任务队列，包含信息如下

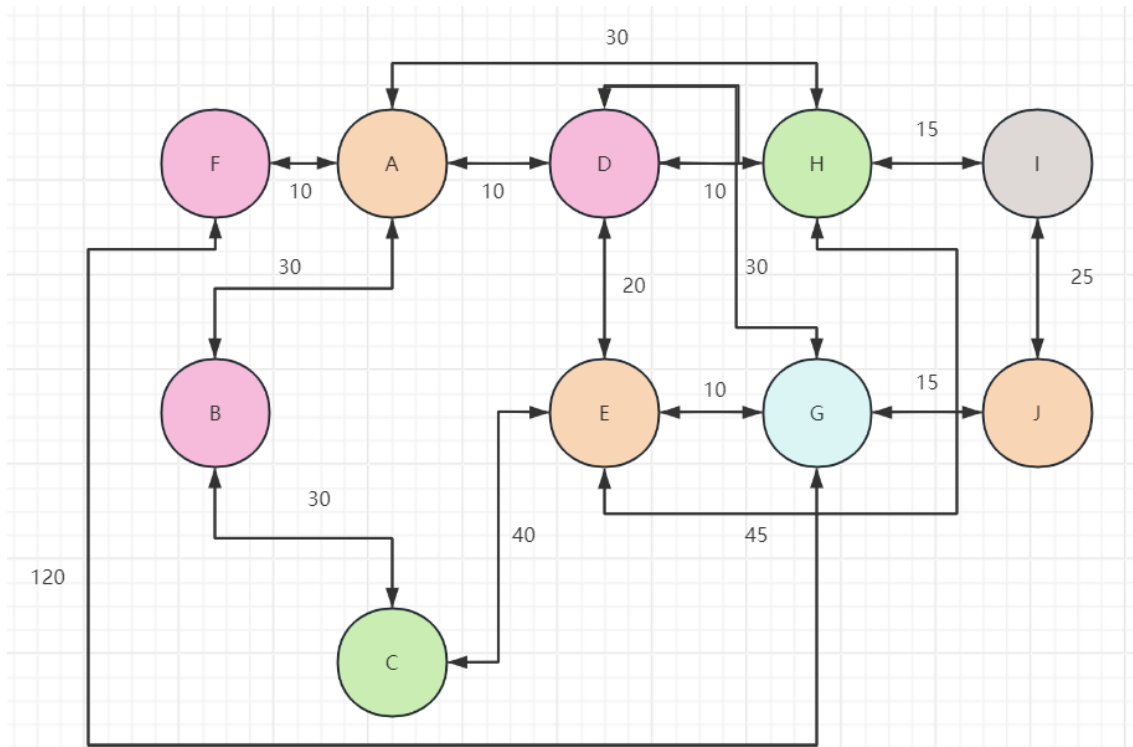
- 包裹编号 (\$1,2,3,...\$)
- 包裹重量 (\$w_1,w_2,w_3,...\$)
- 包裹送达时限 (\$t_1,t_2,t_3...\$)
- 包裹目的地 (名称: A, B, C...)
- 包裹优先级 (1, 2, 3三种值, 值越高, 则代表越需要优先配送)

包裹按照到达先后时间排队，管理员可以手工修改包裹优先级。

2. 还要维护必要的快递小车信息，如下

- 快递小车速度 \$v\$
- 快递小车的自重 \$w_{car}\$
- 快递小车的装载重量上限 \$w_{lim}\$

3. 导入校园地图，本项目地图是静态的，一旦导入不需要修改。其中点 \$A\$ 是菜鸟驿站的位置



4. 根据包裹队列和地图，生成快递无人小车的装载方案和行驶路线，计算送货成本。

送货成本定义如下

$$cost = \sum_{趟数} \sum_{路程段} \text{本段路程长度} \times (w_{car} + \text{此段路程上的包裹总重})$$

注意：路程段包括小车从驿站出发送货后回到驿站的所有路程

项目评分要点

1. 根据包裹队列信息、小车信息和静态地图，指定至少两种不同的送货规划，对同一个测试数据，可以分别运行并计算出结果。对于每种送货规划，给出**计算的送货成本**，并记录有多少包裹未按送达时限送达（70分）

附件中是一个最简单的送货规划的运行示例，供参考。

2. 在**送货过程中加入管理员修改包裹优先级**的操作（模拟被催送货后包裹优先级提高插入到队列最前端），完成送货规划，并计算送货成本，同时记录有多少包裹未按送达时限送达（自己输入）。（10分）

一种简单的实现方式如下：

```
//修改包裹优先级  
修改时刻 修改包裹编号 修改后的优先级
```

3. **增加送达时间的考虑**，要求尽量将包裹在时限内送完，规划运送方式并计算成本。（20分）

最直观的想法是贪心，不过这并非最优解，聪明的同学们不妨多想想为什么？

4. 在整个算法层面上，可能并没有一个策略可以保证任何的属于数据都得到最优解，有兴趣的同学可以挑战一下自己多生成一些测试数据比较不同的策略的成本，一份有效的测试数据（可以反映出不同策略之间的差异）可以获得额外的5分加分，当然，整个项目得分上限依然是100分。
- 5.

项目提供数据

- map_edge.csv: 包含了节点 u ，节点 v ，和 u 到 v 的距离 d ， u 和 v 是双向之间的路径是双向边
- package_1.csv,package_2.csv,...: 包含了快递任务队列信息
- car_1.csv,car_2.csv: 包含了快递小车信息
- test_package.csv: 包含快递任务队列信息，用于编程时测试
- test_car.csv: 包含了快递小车信息，用于编程时测试

样例说明

输入：test_car.csv和test_package.csv

- 小车的速度是10，小车的自重是10，小车的负重上限是30

对这个输入的分析：

- 编号1，A到的时间最少也需要5.5s，所以编号1的订单无法送达，直接不考虑，作为超时快递处理；

在配送完高优先级的快递任务后，剩下的时间也不足以按时完成编号10配送，也作为超时快递处理；

除1之外的其他快递任务均可以有送达的方式

- 编号6，由于他的优先级最高，且可以实现送达，这个包裹可以优先派送
- 对于其他快递任务，可以有多种不同的选择方式

- 按优先级

- 若高优先级的包裹装车后，小车负重上限仍有位置，可以存放低优先级的包裹然后本趟一起送货

如：编号8装上车后，编号2仍能装车

- 若高优先级包裹装车后，不额外考虑小车负重上限，直接送货

如：编号8装上车后，选择开始本趟派送，编号2不装上车

- 相同优先级下

- 什么都不考虑，直接按编号顺序派送

字面意思

- 依照送达目的地，按优先送近的，其次送远的派送（可以利用贪心算法的思想）

跑最短路，计算每个节点距离菜鸟驿站的distance

- 对于同个时间段内，不同方案下超时的快递数量相同，按回到菜鸟驿站时间最早的派送

如编号2,5一起配送和编号3,4一起配送，都会导致另一边超时，但是编号2,5回菜鸟驿站更快，选择配送2,5，让编号3,4超时

- 对于开始配送时，按截止时间最近的派送

对整个快递任务，优先选择\$截止时间-当前时间\$ 小的

- 其他：想象一下真实场景下，你可能会怎么送货？按你的想象模拟一个方案

- 在每一趟配送完后，记录该时间段内的超时快递数量，累加该趟配送的送货成本，

第一轮送编号为6的快递，t=6回到菜鸟驿站，编号1,10超时，即超时快递数量为2
该趟送货成本

$$cost（第一趟）= 30 \times (10 + 30) + 30 \times (10) = 1500$$

在确保所有的快递状态为送达/超时后，输出超时快递总数量和总送货成本

附件

以下给出一个简单的送货策略：

- (1) 按照优先级，将货物排队
- (2) 算驿站到所有其他顶点的最短路，并换算成时间
- (3) 开始送货：

判断当前货物是否会超时，会超时就忽略此货物（相当于丢弃这个任务），不会超时就送货。每次只送一个货物，送达后小车返回驿站再送下一个。

按照这个策略，测试样例的送货结果是：

①计算最短路，得到

- A 0
- B 30
- C 60
- D 10
- E 30
- F 10
- G 40
- H 20

I 35

J 55

②按小车的速度 $v=10$,转换成时间

A 0

B 3

C 6

D 1

E 3

F 1

G 4

H 2

I 3.5

J 5.5

③对快递任务进行筛选, 筛掉一开始就超时或者超出装载上限30的快递任务, 剩余的任务为
id,weight,limit_t,destination,prior

2,10,18,J,1

3,15,13,F,1

4,15,27,G,1

5,20,17,I,1

6,30,17,B,3

7,10,40,J,1

8,20,10,H,2

9,20,9,E,2

10,10,6,J,1

④对快递任务进行排序, 先按照优先级排序, 然后按照默认排序
id,weight,limit_t,destination,prior

6,30,17,B,3

8,20,10,H,2

9,20,9,E,2

2,10,18,J,1

3,15,13,F,1

4,15,27,G,1

5,20,17,I,1

7,10,40,J,1

10,10,6,J,1

⑤配送快递任务:

6,30,17,B,3 判断——可以送达

花费3s运输过去, 花费3s回来, 该轮运输成本= $30 \times (10+30) + 30 \times (10) = 1500$, 此时时间为6s

8,20,10,H,2 判断——可以送达

花费2s运输过去, 花费2s回来, 该轮运输成本= $20 \times (10+20) + 20 \times (10) = 800$, 此时时间为10s

9,20,9,E,2 判断——无法送达

2,10,18,J,1 判断——可以送达

花费5.5s运输过去, 花费5.5s回来, 该轮运输成本= $55 \times (10+10) + 55 \times (10) = 1650$, 此时时间为21s

3,15,13,F,1 判断——无法送达

4,15,27,G,1 判断——可以送达

花费4s运输过去, 花费4s回来, 该轮运输成本= $40 \times (10+15) + 40 \times (10) = 1400$, 此时时间为29s

5,20,17,I,1 判断——无法送达

7,10,40,J,1 判断——可以送达

花费5.5s运输过去, 花费5.5s回来, 该轮运输成本= $55 \times (10+10) + 55 \times (10) = 1650$, 此时时间为40s

10,10,6,J,1 判断——无法送达

总运输成本= $1500+800+1650+1400+1650=7000$

所以最终送货结果为

超时快递:1,3,5,9,10

总运输成本为:7000