

- 1.数据文件说明
 - 1.1 aps_order.csv
 - 1.2 aps_semi_product.csv
 - 1.3 aps_process.csv
 - 1.4 aps_resource.csv
 - 1.5 aps_process_resource.csv
- 2.算法要求说明
 - 2.1 约束条件
 - 2.2 优化目标
 - 2.3 算法输出物
- 3.平台功能要求说明

1.数据文件说明

分为五个文件，分别用以描述订单、商品、半成品工序、资源、工序对应所需的资源。

其中为了简化问题有以下假设「若在算法实现中考虑其中约束部分为加分项」：

- 工人及资源一天只工作8小时，不考虑人员的倒班；
- 工序执行时间只与工序相关，与使用的资源无关；
- 资源之间无依赖关系，如A设备的使用必须需要B这个人；
- 设备不设置利用率，所有设备的利用率为100%；

1.1 aps_order.csv

该文件是描述订单所包含的商品ID、商品数量、订单最早开始时间、订单最晚结束时间等描述订单详情内容，一行数据为某一订单下所需的某一商品信息。

- `order_id`：订单编号；
- `description`：订单描述；
- `product_id`：商品ID，描述订单下的某一商品；
- `product_amount`：商品数量，该订单下该商品所需要的数量；
- `earliest_start_time`：该笔订单的最早开始时间；
- `latest_end_time`：该笔订单的最晚结束时间，订单中的所有商品需在这个时间点之前全部生产完成；

1.2 aps_semi_product.csv

该文件是描述一个商品是由哪些半成品组成的，其中与**商品ID**同名的**半成品**用以标示装配步骤，一行数据为某一商品下所需的某一半成品信息。

- `product_id`：商品ID编号；
- `semi_product_id`：商品所关联的半成品ID

1.3 aps_process.csv

该文件是描述一个半成品下的所有工序，其中以商品ID命名的半成品为装配工序，半成品与工序之间是一对多的关系，一行数据为某一个半成品所需的某一道工序信息。

- `semi_product_id`：半成品ID，用以区别半成品，唯一；
- `semi_product_name`：半成品名称；
- `part_id`：半成品所需要的工序ID，唯一；
- `part_name`：半成品所需要的工序名称，唯一；
- `prev_part_id`：该工序所依赖的前置工序，必须前置工序完成后才可以进行这一道工序，若为空，则不需要前置工序；
- `input_semi_finished_product_id`：该工序所需要的前置半成品ID，以「,」隔开；
- `input_semi_finished_product_amount`：该工序所需要的前置半成品分别的数量，以「,」隔开，次序与前一字段对应；
- `output_semi_finished_product_id`：该道工序结束后所输出的半成品ID，标记该道工序为该半成品的最后一道工序，若该字段为空，则其还有后置工序；
- `production_mode`：该道工序的工艺，分为：BP(批处理)，SP(流处理)。
 - BP：批处理的意思为一次可以进行多数量的执行，可以同时加工 1 至 `maximum_production_quantity` 数量半成品进行该工序，耗时为「`productin_time`」，若干个半成品可以并行。
 - SP：流处理的意思为一次只可以进行一次，多个半成品串行执行。
 - 例：`production_mode` = BP，`maximum_production_quantity` =10，`productin_time` = 15
该工序可以并行着做，可以同时针对1-10个半成品进行加工该道工序，无论是1个还是10个半成品消耗时间都是15分钟。
例：`production_mode` = SP，`maximum_production_quantity` =1，`productin_time` = 5
该工序只能串行着做，一次执行至多1个半成品，执行时间为5分钟。
- `production_time`：该道工序的执行时间，以分钟为单位；
- `minimum_production_quantity`：最小可执行数量；
- `maximum_production_quantity`：最大可执行数量；
- `workspace`：该工序属于什么生产车间，该工序只可以使用该生产车间下的资源

1.4 aps_resource.csv

该文件是描述资源的相关信息。

- `resource_id`：资源id，唯一；
- `name`：资源名称；
- `resource_type`：资源所属于最大的类别；
- `amount`：同质资源的数量，该数量表示一模一样的资源的总数；

- `resource_attributes`：该资源所属于的所有类别，如 `res_2` 「高级机床，机床」该资源是高级机床，同时它也属于一台机床，当工序需要机床，但机床数量不够时，该资源也可以被使用；
- `basic_attribute`：资源所属的最高级的类别，如 `res_2` 「高级机床，机床」，高级机床是机床的子集，该资源所属于的最小的集合为「高级机床」，故 `basic_attribute` 为高级机床；
- `workspace`：该资源所属于的生产车间。

1.5 aps_process_resource.csv

该文件是描述工序与其所需的资源之间的匹配的关系，工序与资源之间是一对多的关系。

- `part_id`：工序ID与 `aps_process` 中的 `part_id` 对应；
- `name`：工序名称；
- `resource_attributes`：该工序所需要的资源属性，与 `aps_resource` 中的 `resource_attributes` 相关；
- `amount`：数量，该工序需要这类资源的数量；

2.算法要求说明

2.1 约束条件

- 满足基础的约束，如工序之间前后置关系、最晚交货时间、同一时刻资源的上限等。
- 在满足基础约束的基础上，可以自定义添加认为可以考虑的因素作为加分项，包括但不限于打破本题所做的假设。

2.2 优化目标

- 总工时最短。
- 计算时间最短。

2.3 算法输出物

- 输出的结果需要具体到某一个具体的资源，即需要对每个任务进行具体资源ID的分配。

3.平台功能要求说明

包括但不限于以下要求：

- 订单，资源，工序的增删改查，导入导出功能。
- 插单问题的动态重排策略。