# 题目：连续资源块分配

## 概念

**资源块**： resource block(RB)资源块是连续的，每个RB只能被一个用户使用;

**资源块的编号**：总计N个连续资源块，编号为0 ~ N-1;

**用户请求**：用户需要使用连续的RB段。用户请求信息 包括数量、弹性 和 位置偏好。

**数量** < N；

**弹性指**：Fully， Partial

请求数量被完全满足(Fully)：当不存在满足需求的连续可用RB段时，分配失败；

请求数量部分满足(Partial)：当不存在完全满足需求的连续可用RB段时，分配可用的最长RB段；当最长可用RB段为0时，分配失败；

**位置偏好包括**：Forward， Reverse，SpecifyPoint， SpecifyRange

正向(Forward)，从0 号RB开始正向寻找满足要求的RB段；

逆向(Reverse)，从N-1->0 逆向寻找；

指定位置开始(SpecifyPoint)：用户指定起始位置，正向寻找；

指定区域(SpecifyRange): 在用户指定区域，正向寻找；

## 要求

1. 支持多个用户分配N个RB块，多个用户的请求按顺序处理；
2. 支持不同 的分配方式。 一种分配方式指 特定的 弹性 + 位置偏好 的组合，如下述两种分配方式：

Fully-Forward 方式 (正向搜索，只接受完全满足的)

Partial-SpecifyRange 方式 (指定区域内接受部分满足)

1. 支持不同分配方式间顺序执行，如选择三种分配方式{Partial-SpecifyRange, Fully-SpecifyPoint, Fully-Forward}顺序执行, 任意一种成功即停止分配。
2. 当存在多个长度相同的备选分配结果(段)时，选择搜索方向上的第一个段。(即正向寻找时，段起始位置尽可能小；逆向搜索时，段起始位置尽可能大)。

## **用例**

请求的格式如下：

{需求量， <{分配方式, 分配参数}>}，

<>表示有多个元素。

分配参数中point\range 均是闭区间。

### eg1:

100RB， 请求序列为：

{10, {Partial-SpecifyRange, range = [40, 59] }};

{45, {Fully-SpecifyPoint, point = 66}, {Fully-Forward}};

{20, {Fully-Reverse}};

{25, {Fully-Forward}}

分配结果是{[40, 49], [50, 94], [20,39], Fail}

### eg2:

270RB, 请求序列为:

{20, {Partial-SpecifyRange, range = [70, 99] }};

{10, {Partial-SpecifyRange, range = [140, 179] }};

{10, {Partial-SpecifyRange, range = [210, 239] }};

{20, {Full-SpecifyRange, range = [80, 99],

{Fully-SpecifyPoint, point=160},

{Fully-Forward} };

{80, {Fully-SpecifyPoint, point = 100}

{Fully-Reverse}};

{100, {Partial-Forward}}

分配结果是{[70, 89], [140, 149], [210, 219], [160, 179], Fail, [0, 69]}

## **编码要求**

1. 自行定义合适的接口；
2. 用例可以通过；
3. 考虑函数的单一职责；
4. 考虑处理效率；