Problem background

In 5G we have two very powerful signal transmission features, which are beamforming and Multi-User signal transmission (MU MIMO). Typically, 5G base station is equipped with 64 element rectangular antenna (you may think of it as a square grid with 8x8 elements, but in reality, it is slightly more complicated). For downlink transmissions (from base station to users), in many cases, phase control at antenna elements allows to concentrate energy in direction of the user, increasing signal for him, and wasting less energy in other directions. Moreover, using slightly more advanced math about phases for antenna elements, given some conditions of user independence, it is possible to *focus* the first signal onto the first user, and *focus* the second signal of the same frequency onto the second user (in different direction), and simultaneously achieve minimum of the first signal in direction of second user (and second signal in the first direction too). This allows to concentrate energy in required directions, and also serve several users using same frequency resources.

In uplink (transmission direction from users to base station), base station capabilities have some differences. In this case it is not focusing signals, but rather separating them. If user sends a signal to the base station, each antenna element of base station will receive almost the same copy of user signal, with some phase differences between antenna elements. Phase profile depends on a position of a user relative to base station antenna. For multi-user transmissions in uplink (signals are transmitted from several users to base station), base station is able to receive and extract individual signals of several users, if conditions of channel independence in phase space(=orthogonality) hold. In our problem we have Beam id as a user attribute, which corresponds to one of directions of user in antenna coordinate system. Same beam of several users means signal of those users come from approximately same direction, so all received signals spoil each other, and thus cannot be received simultaneously on the same frequency. If beams of some users are different, their signals do not interfere with each other, and their channels are considered to be orthogonal (independent). In this latter case, base station is able to receive and extract signals of such users, even if they transmit simultaneously on the same frequency resource.

Dictionary of 5G

- **RB** Resource block, the smallest range of frequencies, which can be given to a user.
- **Beam** Direction of a signal between base station and UE.
- **Bandwidth** Total range of frequencies. It is divided into RBs.
- **Base station**. Array of antennas with modulator controlled by a computer.

Problem statement

Your task is to develop a scheduler of data transmission from N users $0, 1, ..., N-1$ to a base station. Each user i needs a_i RBs and belongs to beam b_i. For data transmission the base station have a bandwidth consisting of M RBs $\mathcal{B} = [0, 1, ..., M)$, but some of RBs are reserved and can't be used. More precisely, there are K RB interval $\mathcal{R}_0 = [s_0, e_0)$ (beginning of interval including, its end not including), $\mathcal{R}_1 = [s_1, e_1)$, ..., $\mathcal{R}_K = [s_K, e_K)$ of reserved RBs.

Scheduler should work as follows. It selects no more than J disjoint intervals of not reserved RBs $\mathcal{I}_0, \mathcal{I}_1, ..., \mathcal{I}_J \subset \mathcal{B} \setminus (\cup_{i=0}^{K} \mathcal{R}_i)$. These intervals should be distributed among users. Denote the set of users receiving interval $\mathcal{I}_j$, by $\mathcal{U}_j \subset \{0, ..., N-1\}$. The scheduling must satisfy the following rules:

1. An RB interval $\mathcal{I}_j$ might be given to no more than L users ($|\mathcal{U}_j| \leq L$). It is made avoid the risk of high interference between users.
2. An RB interval $\mathcal{I}_j$ can not be given to a pair of users u and v from the same beam, because their signals will interfere ($u, v \in \mathcal{U}_j \Rightarrow b_u \neq b_v$).
3. A cell phone can transmit signal only on continuous frequency range. Thus, the union of the intervals given to a user u should also be an interval $\cup_j^{u \in \mathcal{U}_j} = [a, b)$ for some $0 \leq a, b \leq M$. In particular, there are no reserved RBs between two intervals given to one user.

Result of work of the scheduler will be estimated by total number of RBs given to users. A user u may receive less RBs than he needs (number a_u). On the other hand, a user u can receive more than a_u RBs, but all extra RBs (starting $a_u + 1$) will bring no credits to the scheduler.

INPUT:

$20 \le N \le 128$ the number of UEs, $0 < a_u \le M$ number of RBs UE u needs, $0 \le b_u < 32$ beam of UE i.

$64 \le M \le 512$ the total number of RBs in the bandwidth, $0 \le K \le 4$ number of interval of reserved RBs, $\mathcal{R}_i$ reserved RBs.

$0 < J \le 16$ maximal number of interval of the schedule $\mathcal{I}_0, ...,$ L maximal number of users in the interval $\mathcal{I}_j$.

OUTPUT:

Intervals $\mathcal{I}_0, \mathcal{I}_1, ..., \mathcal{I}_J$ with corresponding sets of users $\mathcal{U}_0, \mathcal{U}_1, ..., \mathcal{U}_J$.

CONSTRAINTS:

1. $\forall i < J$ and $\forall k < K$,: $\mathcal{I}_j \cap \mathcal{R}_k = \varnothing$;
2. $\forall i, j < J$, $i \ne j$: $\mathcal{I}_j \cap \mathcal{I}_i = \varnothing$;
3. $\forall j < J$: $|\mathcal{U}_j| \le L$;
4. $\forall j < J$, $\forall u, v \in \mathcal{U}_j$, $u \ne v$: $b_u \ne b_v$;
5. $\forall u < N$: $\exists\, 0 \le a, b \le M$ s.t. $\cup_j^{u \in \mathcal{U}_j} \mathcal{I}_j = [a, b)$.

TARGET:

$$\sum_{u=0}^{u<N} \min(a_u, \sum_j^{u \in \mathcal{U}_j} \text{length}(\mathcal{I}_j)) \to \max$$

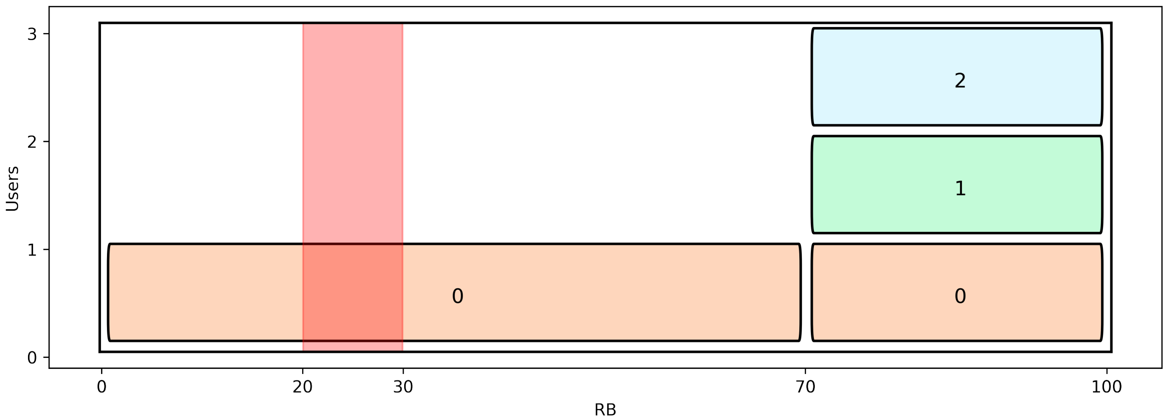
Visual examples

Let $N = 5$, $a_0 = 100$, $a_1 = a_1 = a_2 = a_3 = a_4 = 30$, $b_0 = 0$, $b_0 = 0$, $b_1 = 1$, $b_2 = 2$, $b_3 = 0$ and $b_4 = 4$ (we mark beam of a user by its color).
$M = 100$ (that means $\mathcal{B} = \{0, 1, ..., 99\}$, $K = 1$, $\mathcal{R}_0 = [20, 30)$, $J = 2$ and $L = 3$.

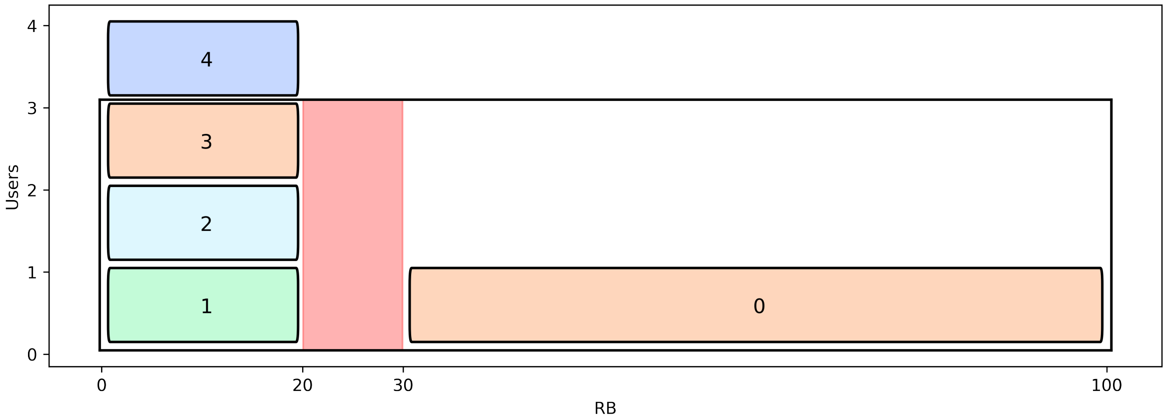
Incorrect solutions

Consider several examples of incorrect solutions.

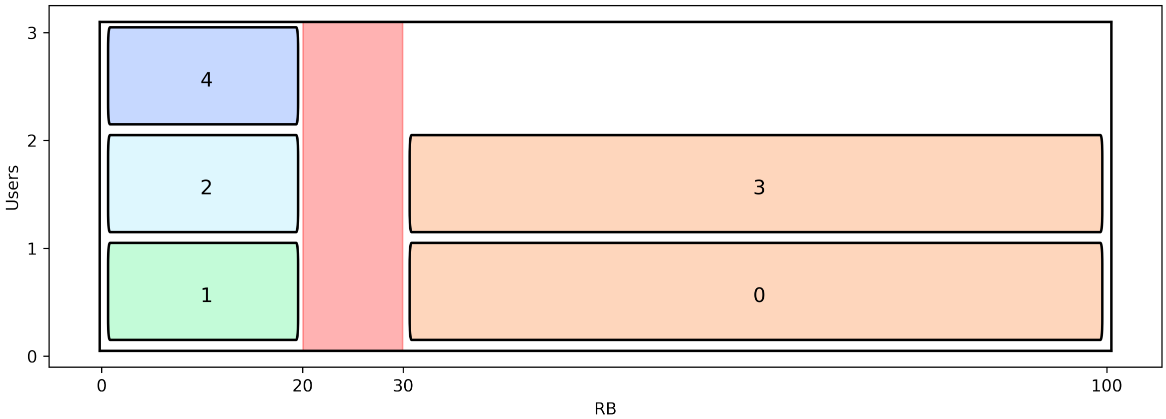
1. $\mathcal{I}_0 = [0, 70)$, $\mathcal{U}_0 = \{0\}$, $\mathcal{I}_1 = [70, 100)$, $\mathcal{U}_1 = \{0, 1, 2\}$.
   Violation of constraint 1. Interval $\mathcal{I}_0$ contains reserved RBs.



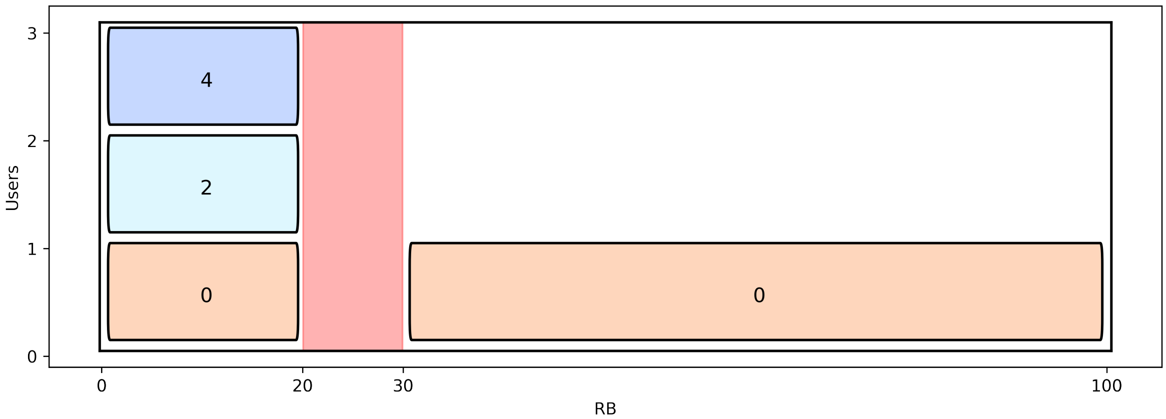
2. $\mathcal{I}_0 = [0, 20)$, $\mathcal{U}_0 = \{1, 2, 3, 4\}$, $\mathcal{I}_1 = [30, 100)$, $\mathcal{U}_1 = \{0\}$. Violation of constraint 3.
Interval $\mathcal{I}_0 = [0, 20)$ is given to $4 > L = 3$ users.



3. $\mathcal{I}_0 = [0, 20)$, $\mathcal{U}_0 = \{1, 2, 4\}$, $\mathcal{I}_1 = [30, 100)$, $\mathcal{U}_1 = \{0, 3\}$.
Violation of constraint 4. Interval $\mathcal{I}_1$ is given to two users with beam $b_0 = b_3 = 0$.



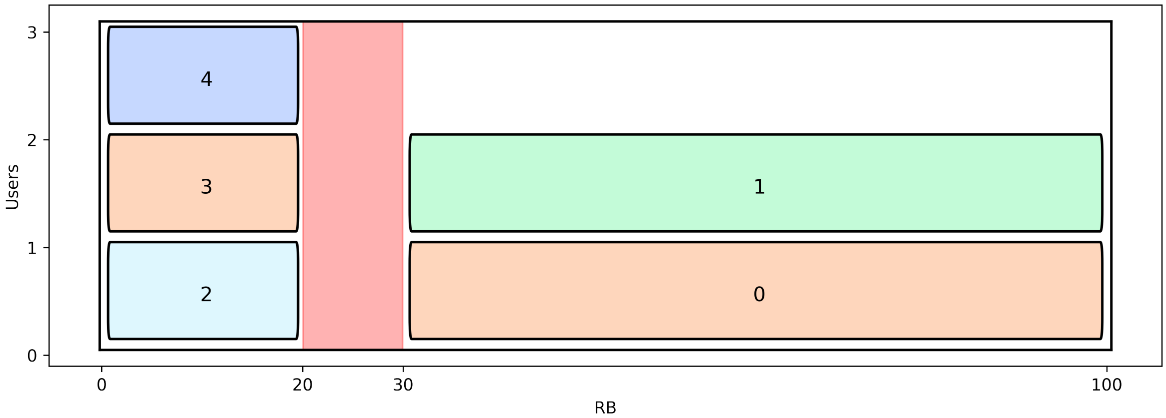
4. $\mathcal{I}_0 = [0, 20)$, $\mathcal{U}_0 = \{0, 2, 4\}$, $\mathcal{I}_1 = [30, 100)$, $\mathcal{U}_1 = \{0\}$.
Violation of constraint 5. User 0 received 2 intervals $\mathcal{I}_0 = [0, 20)$ and $\mathcal{I}_1 = [30, 100)$. There is a gap between them!



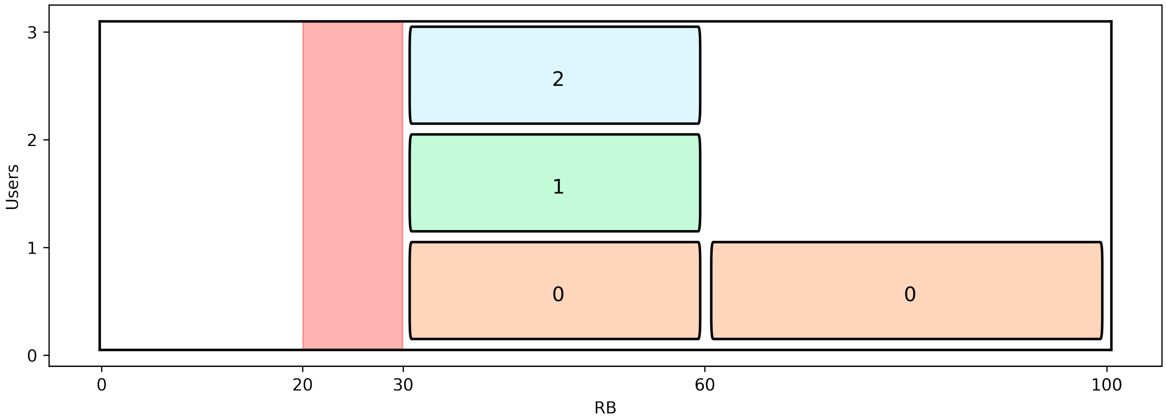
Correct solutions

Let us consider two correct solutions and compute their scores.

1. $\mathcal{I}_0 = [0, 20)$, $\mathcal{U}_0 = \{2, 3, 4\}$, $\mathcal{I}_1 = [30, 100)$, $\mathcal{U}_1 = \{0, 1\}$.
   User 0 receives 70 RBs, user 1 receives 30 RBs, users 2,3 and 4 receive 20 RBs. Total score is 160 RBs.



2. $\mathcal{I}_0 = [30, 60)$, $\mathcal{U}_0 = \{0, 1, 2\}$, $\mathcal{I}_1 = [60, 100)$, $\mathcal{U}_1 = \{0\}$.
   User 0 receives 70 RBs, user 1 receives 30 RBs, users 1 and 2 receive 30 RBs. Total score is 130 RBs.



Remarks

- A user may transmit less RBs than he needs or even transmit nothing.
- A user u may transmit more RBs than he needs but in the case he will add only a_u to the score.
- A user can be represented in several intervals, but all these intervals must form a bigger interval. In other words, if we sort these intervals from left to right, $\mathcal{I}_{i+1}.start = \mathcal{I}_i.end$. See user 0 in correct solution 2.
- Number of intervals must be not greater than J.

Implementation, input/output

You need to implement a function

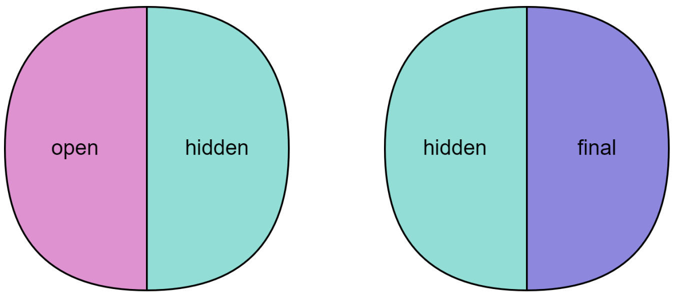
```
std::vector<Interval> Solver(int N, int M, int K, int J, int L,
                             std::vector<Interval> reservedRBs,\\vector users is empty.
                             std::vector<UserInfo> userInfos)
```

where:

- ```
  struct UserInfo {
      int rbNeed;
      int beam;
      int id;
  };
  ```
- ```
  struct Interval {
      int start, end;
      std::vector<int> users;
  };
  ```
- `N` – the number of users;
- `M` – the number of RBs;
- `K` – the number of reserved intervals;
- `J` – the number of intervals with users;
- `L` – the maximal number of users on interval.

Testing features

There are 3 sets of tests: *open*, *hidden*, *final*.



During the competition, your solution will be evaluated as an average over the open and the hidden sets. For the final score we will use the hidden and final sets. Sets are homogeneous (they use same distributions, lower/upper-limits, etc.), but are not equivalent. The open set is open for all participants for research (consider that function Solver is focused on solving only one problem, but open set has multiple test-cases in one file).

Competition rules

- The number of submissions is not limited;
- You may participate both alone and in team (up to 3 persons);
- Solution must follow the C++17 standard;
- Time limit for one given file (2000 test-cases) is 1 second;
- Memory limit for one given file (2000 test-cases) is 256 MB.

Open set format

Given open set `open.txt` has the following format.
The first string is the number of the problems in the set. Next for each problem we have.

```
N M K J L
```

Next we have `K` strings for each reserved interval i:

$\mathcal{R}_i.begin\ \mathcal{R}_i.end$

After there are `N` strings for each user u:

$a_u\ b_u$