K: Knavish Knockout

Problem author: Michael Ruderer

Problem

Input: $n \cdot k$ apples with distinct sizes and poison values

Task:

- Partition the apples into k groups of size n.
- Maximize total consumed poisoned, when the smallest apple in each group is eaten.

K: Knavish Knockout

Problem author: Michael Ruderer

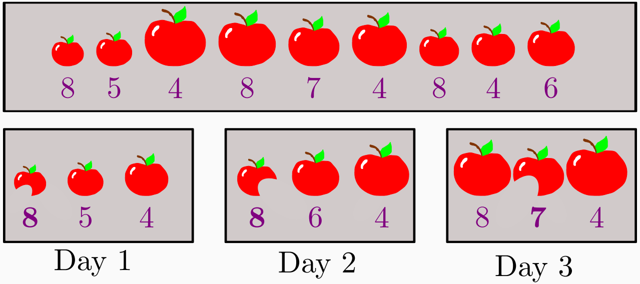
Problem

Input: $n \cdot k$ apples with distinct sizes and poison values

Task:

- Partition the apples into k groups of size n.
- Maximize total consumed poisoned, when the smallest apple in each group is eaten.

E.g. Sample 2 with $n = k = 3$:



K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

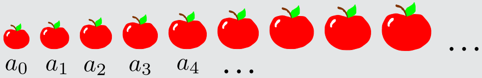
- Let $a_0, a_1, \ldots, a_{n \cdot k-1}$ be the apples sorted by size, smallest to largest

K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

- Let $a_0, a_1, \ldots, a_{n \cdot k-1}$ be the apples sorted by size, smallest to largest

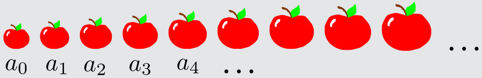


K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

- Let $a_0, a_1, \ldots, a_{n \cdot k - 1}$ be the apples sorted by size, smallest to largest



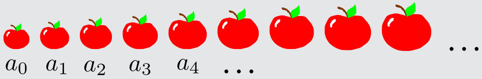
- First Observation: In any solution a_0 has to be eaten

K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

- Let $a_0, a_1, \ldots, a_{n \cdot k-1}$ be the apples sorted by size, smallest to largest



- First Observation: In any solution a_0 has to be eaten
- In general: Among $a_0, \ldots, a_{n \cdot i}$, at least $i+1$ apples have to be eaten (Pigdeonhole-like argument)
  This condition is sufficient to construct a valid solution!
  Any set E of k apples is the set of apples eaten in a feasible solution if and only if

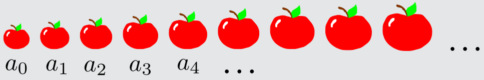
$$|E \cap \{a_0, \ldots, a_{n \cdot i}\}| \geq i + 1 \text{ for all } i = 0 \ldots k-1$$

K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

- Let $a_0, a_1, \ldots, a_{n\cdot k-1}$ be the apples sorted by size, smallest to largest



- First Observation: In any solution a_0 has to be eaten
- In general: Among $a_0, \ldots, a_{n\cdot i}$, at least $i+1$ apples have to be eaten (Pigdeonhole-like argument)
  This condition is sufficient to construct a valid solution!
  Any set E of k apples is the set of apples eaten in a feasible solution if and only if

$$|E \cap \{a_0, \ldots, a_{n\cdot i}\}| \geq i+1 \text{ for all } i = 0 \ldots k-1$$

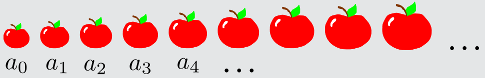
Algorithm: Greedily construct optimal set E satisfying this condition.
For $i = 0, \ldots, k-1$, add the most poisonous apple from $\{a_0, \ldots, a_{n\cdot i}\} \setminus E$ to E.

K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

- Let $a_0, a_1, \dots, a_{n \cdot k-1}$ be the apples sorted by size, smallest to largest



- First Observation: In any solution a_0 has to be eaten
- In general: Among $a_0, \dots, a_{n \cdot i}$, at least $i+1$ apples have to be eaten (Pigdeonhole-like argument)
  This condition is sufficient to construct a valid solution!
  Any set E of k apples is the set of apples eaten in a feasible solution if and only if

$$|E \cap \{a_0, \dots, a_{n \cdot i}\}| \geq i+1 \text{ for all } i = 0 \dots k-1$$

Algorithm: Greedily construct optimal set E satisfying this condition.
For $i = 0, \dots, k-1$, add the most poisonous apple from $\{a_0, \dots, a_{n \cdot i}\} \setminus E$ to E.

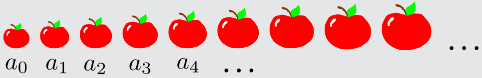
Implementation: Use a max-heap to efficiently find the most poisonous apple in each step.

K: Knavish Knockout

Problem author: Michael Ruderer

Solution Idea

- Let $a_0, a_1, \ldots, a_{n \cdot k-1}$ be the apples sorted by size, smallest to largest



- First Observation: In any solution a_0 has to be eaten
- In general: Among $a_0, \ldots, a_{n \cdot i}$, at least $i+1$ apples have to be eaten (Pigdeonhole-like argument)
  This condition is sufficient to construct a valid solution!
  Any set E of k apples is the set of apples eaten in a feasible solution if and only if

$$|E \cap \{a_0, \ldots, a_{n \cdot i}\}| \geq i+1 \text{ for all } i = 0 \ldots k-1$$

Algorithm: Greedily construct optimal set E satisfying this condition.
For $i = 0, \ldots, k-1$, add the most poisonous apple from $\{a_0, \ldots, a_{n \cdot i}\} \setminus E$ to E.

Implementation: Use a max-heap to efficiently find the most poisonous apple in each step.

Complexity: $O(n \cdot k \cdot \log(n \cdot k))$