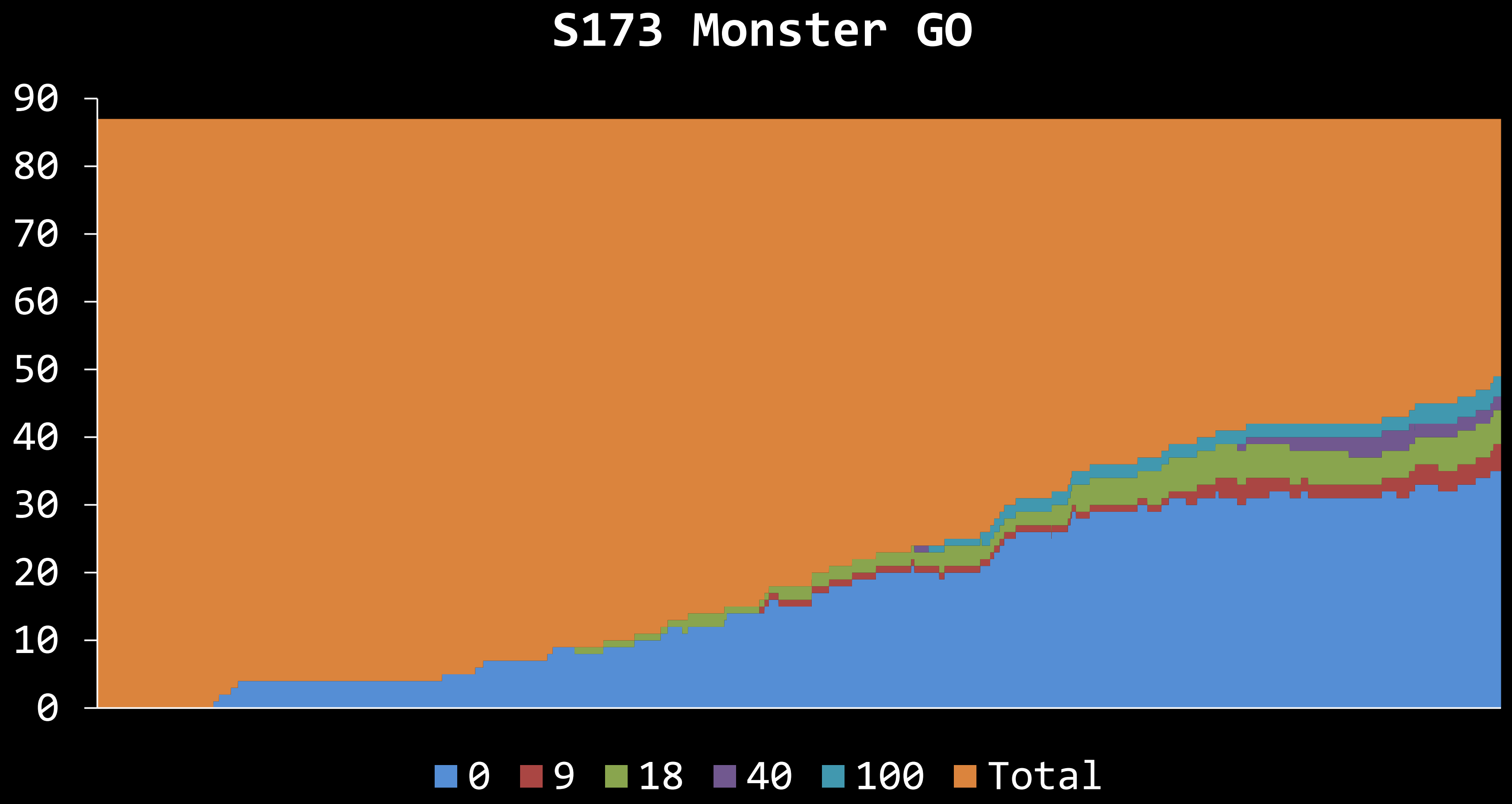
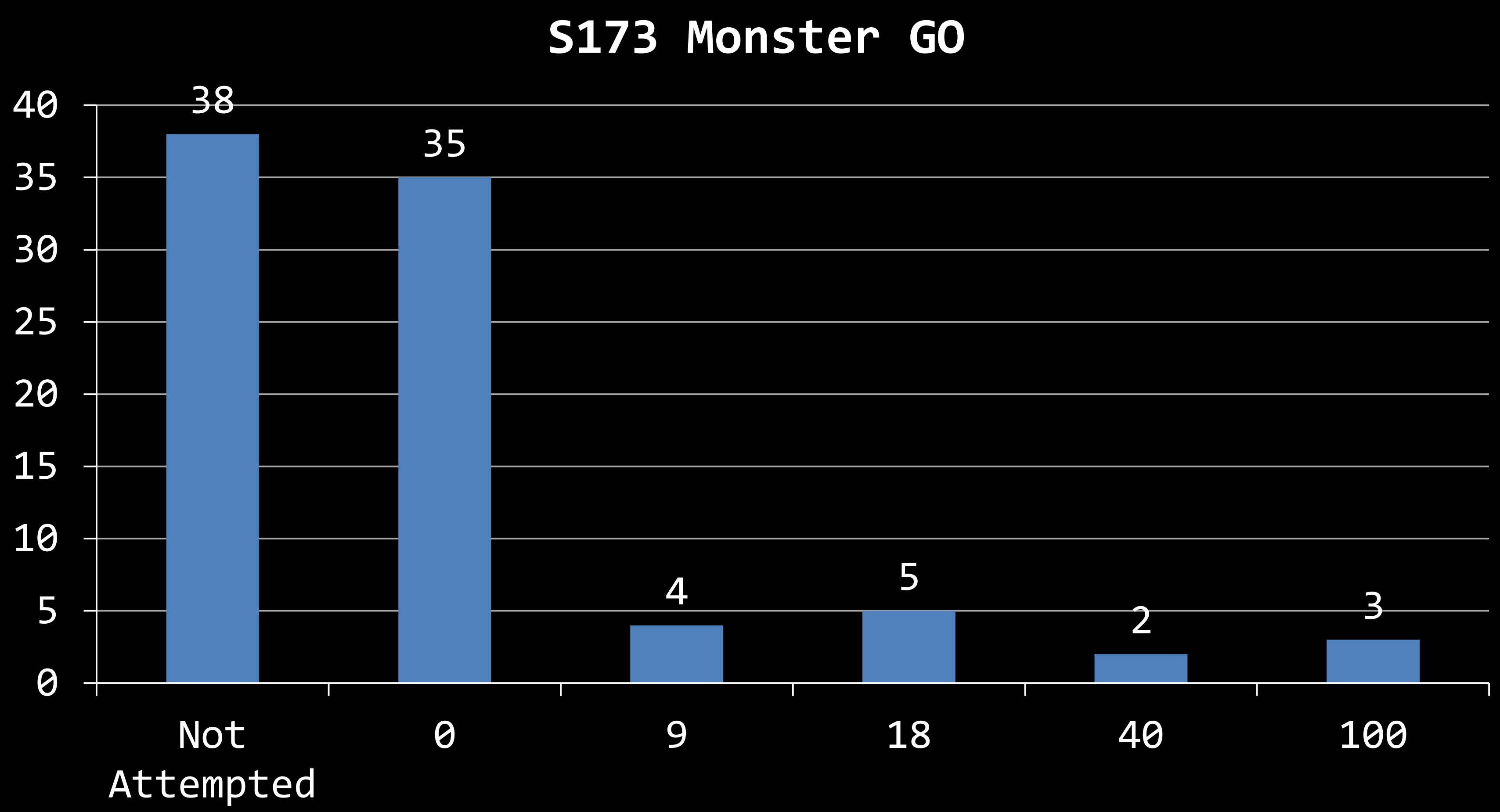
S173 Monster GO

Author: Anson Ho

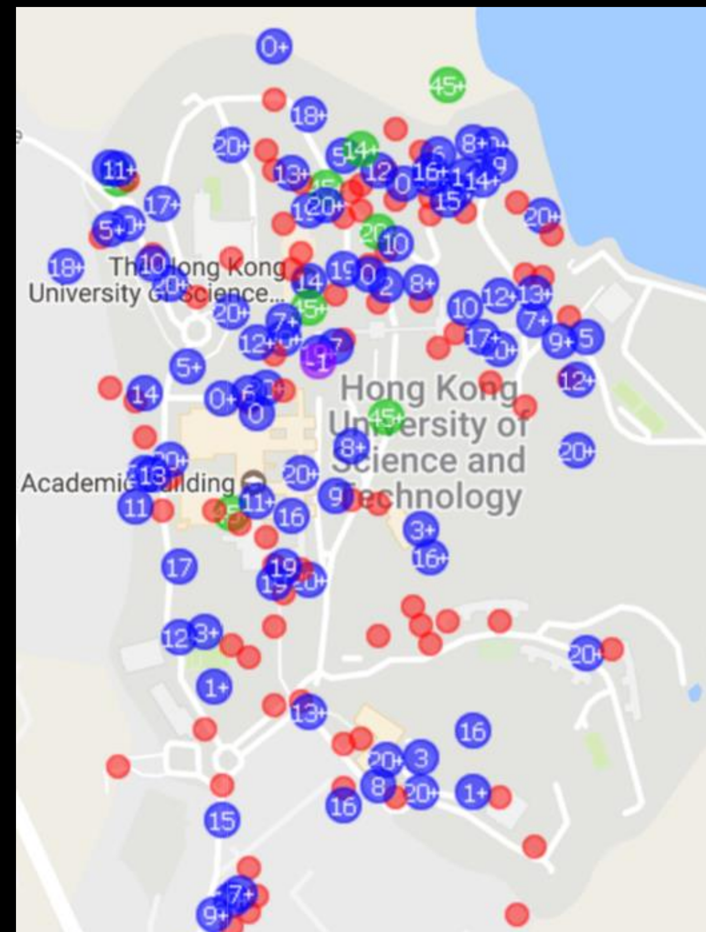
Statistic



Statistic


S173 Monster GO
40
35
30
25
20
15
10
5
0
38
35
4
5
2
3
Not Attempted
0
9
18
40
100

Background



https://cdn.igeeksblog.com/wp-content/uploads/2016/08/Find-Nearby-Pokemon-with-Sightings-Screen-in-Pokemon-Go.jpg

https://www.tqgames.com/

Problem Statement

- N monsters live in N caves
- Different monsters live in different caves

Cave	1	2	3	4
Monster	C	A	B	D

Problem Statement

- Given a list of M radars
- Each radar have a given set of caves in range
- Each radar can show the monsters in the caves in range in an arbitrary order

Problem Statement

Cave	1	2	3	4
Monster	C	A	B	D

Radar	Cave 2	Cave 3	Cave 4

- {A, B, D}
- {D, A, B}
- {B, A, D}

- {A, B}
- {C, A, B, D}
- {B, C, D}

Problem Statement

- Output the minimum number K such that the first K radars can determine the location of each monster

- ⇔ there is exactly one permutation of monster locations corresponding to the information of the K radars

Example

- N = 2, M = 2
- Range of radar 1 = {cave 1, cave 2}
- Range of radar 2 = {cave 2}

- Ans: K = 2

- Radar 1 actually gives no useful information
- The monster in cave 1 can be determined by method of elimination once the monster in cave 2 is known

Subtask 1

- N, M <= 2

- When N = 1
- No need radar
- Output 0
- Monster 1 always lives in cave 1

Subtask 1

- N, M <= 2

- When N = 2
- Possible ranges of radars
- {1}, {2}, {1, 2} = {2, 1}
- $3 + 3^2$ (or $4 + 4^2$) possibilities
- Hardcode

Subtask 2

- N <= 2, NM <= 2e5

- When N = 2
- Possible ranges of radars
- {1}, {2}, {1, 2} = {2, 1}
- Done ⬄ {1} or {2} appear

Observation

- Cave i and Cave j are distinguishable
  ⇔ you can partition N monsters into two groups such that the monsters in cave i and cave j are in different groups
  ⇔ the sets of radars in range are different

Observation

- Denote the sets of radars in range by R_i and R_j for cave i and cave j respectively

- If $R_i = R_j$
  Then the monsters in cave i and cave j will be both present or both absent in the result of each radar
  → Not distinguishable

Observation

- If $R_i \neq R_j$
  Then there is one radar such that cave i is in range and cave j is not in range, or in the other way
  → Monsters can be partitioned by the result of that radar
  → Distinguishable

- Thus,
  different set ⇔ distinguishable

Observation

- The radars are enough
  ⇔ the caves are pairwise distinguishable
  ⇔ the sets of radars in range are pairwise different

Subtask 3

- N <= 10, NM <= 2e5

- After a radar is added
- Set the pair of caves to be distinguishable if one is present and one is absent
- Check if every pair of caves is distinguishable

- $O(N^2M)$

Subtask 4

- N, M <= 2e5, NM <= 2e5
- "Not distinguishable" is an equivalence relation
- So the monsters can be partitioned into equivalence classes

Subtask 4

- N, M <= 2e5, NM <= 2e5

- ~~“Not distinguishable” is an equivalence relation~~

- ~~So the monsters can be partitioned into equivalence classes~~

Subtask 4

- N, M <= 2e5, NM <= 2e5
- If A, B are not distinguishable and B, C are not distinguishable
- Then A, C are not distinguishable
- Monsters can be partitioned into groups such that the members are pairwise not distinguishable

Subtask 4

- N, M <= 2e5, NM <= 2e5

- Reduce unnecessary checking
- Recall
  Same set ⇔ not distinguishable
- How to efficiently check the equality of set?

Subtask 4

- For each cave, define a binary number where the i^{th} digit represents whether the cave is in the range of radar i

Subtask 4

- Radar 1 = {1}
- Radar 2 = {1,2}
- Radar 3 = {2}

- Cave 1 = 110_2
- Cave 2 = 011_2
- Cave 3 = 000_2

Subtask 4

- If the N numbers are distinct
- (can be checked by hashing)
- Then it is done

- O(NMlogN)

Subtask 4

- If the N numbers are distinct
- (can be checked by hashing)
- Then it is done

- O(NMlogN)
- Not intended

Subtask 5

- N, M <= 2e5, input size = O(2e5)

- Not to check every N numbers after a radar is added

Subtask 5

- N, M <= 2e5, input size = O(2e5)

- Not to check every N numbers after a radar is added
- Else TLE

Subtask 5

- Method 1

- If the first x radars are enough
- Then the first y (y > x) radars are also enough

- Binary search
- O(NlogNlogM)

Subtask 5

- Method 1

- If the first x radars are enough
- Then the first y (y > x) radars are also enough

- Binary search
- O(NlogNlogM)
- Not intended

Subtask 5

- Method 2

- Notice that a group is split when the members are not
  all in range OR all not in range
  for the newly added radar

Subtask 5

- Method 2

- Implementation

- Define a group id for each group
- Split group → change id
- Only change id of those included in input

Subtask 5

- Method 2

Cave	1	2	3	…	N
Group id	1	1	1	2	K

- Radar = {1, 2}

Cave	1	2	3	…	N
Group id	K + 1	K + 1	1	2	K

Subtask 5

- Method 2

Cave	1	2	3	…	N
Group id	1	1	1	2	K

- Radar = {1, 2, 3}

Cave	1	2	3	…	N
Group id	1	1	1	2	K

Subtask 5

- Method 2

Cave	1	2	3	…	N
Group id	1	1	1	2	K

- Radar = {1, 2, 3}

Cave	1	2	3	…	N
Group id	K + 1	K + 1	K + 1	2	K

Subtask 5

- Method 2

Cave	1	2	3	…	N
Group id	1	1	1	2	K

- Radar = {6, 8, 9}

Cave	1	2	3	…	N
Group id	1	1	1	2	

Subtask 5

- Method 2

Cave	1	2	3	4	…	N
Group id	1	1	2	2	3	K

- Radar = {2, 3}

Cave	1	2	3	4	…	N
Group id	1	K + 1	K + 2	2	3	K

Subtask 5

- Method 2
- O(input size)
- Expected score = 100   :D

Thank you