# J262 - Bowling Score Recovery

Darren Lu {firewater}

2026-02-14

# Background

- Problem idea by hywong1
- Preparation by hywong1
- Presentation by firewater

# Problem Restatement

- You are given a sequence of pin counts for the N bowling balls that did not miss, and you can insert arbitrary amounts of misses in between
- Find the maximum possible score under the given scoring:
  - Strike: 30 points
  - Spare: 10 + (first ball pin count) points
  - Open Frame: (sum of both ball pin counts) points

- Fun fact: this is the scoring system used in the World Bowling Tour!

# Statistics

| Task | Attempts | Max | Mean | Std Dev | Subtasks | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| J262 - Bowling Score Recovery | 97 | 100 | 73.69 | 25.284 | 10: 95 | 13: 94 | 14: 86 | 17: 92 | 18: 62 | 28: 39 |

First solved by **sms28128** at **0h 12m**

# Subtask 1: $N = 2$, $A\_1, A\_2 \neq 10$

- A strike is impossible here
- Check if the two pins form a spare
  - If so, the score is 10+A_1
- Otherwise we can always form open frames with any two adjacent records by inserting 0 as follows:

| | | | |
|---|---|---|---|
| A_1 | 0 | A_2 | 0 |

- Even if A_1, A_2 can form an open frame, we have the score being A_1+A_2 in both cases

# Subtask 2: A_i < 5

- Both strikes and spares are impossible
- We can use a similar strategy to the open frame case from subtask 1:

| | | | | | | |
|---|---|---|---|---|---|---|
| A_1 | 0 | A_2 | 0 | ... | A_N | 0 |

- Hence the score is the sum of all A_i

# Subtask 3: $A_i + A_{(i+1)} \neq 10$

- Spares are impossible
- How do we deal with strikes?
- Since a strike (a 10) gives 30 points, we can just replace every 10 to 30
  - This functionally makes strikes the same as open frames
- We then can just perform the solution from subtask 2

# Subtask 4: N > 1, A_i + A_(i+1) = 10, A_1 > A_2

- Every adjacent pair can form a spare
- The array should look like one of this:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| x | 10-x | x | 10-x | … | x | 10-x | x | (odd N) |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| x | 10-x | x | 10-x | … | x | 10-x | (even N) |

# Subtask 4: N > 1, A_i + A_(i+1) = 10, A_1 > A_2

- For the even N case, consider the following way of grouping entries into spares:

| | | | | | | |
|---|---|---|---|---|---|---|
| x | 10-x | x | 10-x | … | x | 10-x |

- This maximizes the number of spares that we can get

# Subtask 4: N > 1, A_i + A_(i+1) = 10, A_1 > A_2

- For the odd N case, consider both ways of grouping entries into spares:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **x** | 10-x | **x** | 10-x | … | **x** | 10-x | x |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| x | **10-x** | x | **10-x** | … | x | **10-x** | x |

- Since we have floor(N/2) spares in both groupings, we will use the first grouping since the score of each spare will be larger
    - The first grouping has each spare giving 10+x points
    - The second grouping has each spare giving 10+(10-x) points
    - We are given x > 10-x by the constraints

# Subtask 5: A_i + A_(i+1) = 10

- When will our previous solutions fail?
- Consider the following example for the even N case:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **4** | 6 | **4** | 6 | **4** | 6 | **4** | 6 |

- Our previous grouping gives (10+4)*4 = 56 points
- Now consider the following grouping:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 4 | **6** | 4 | **6** | 4 | **6** | 4 | 6 |

- This gives 4+(10+6)*3+6 = 58 points

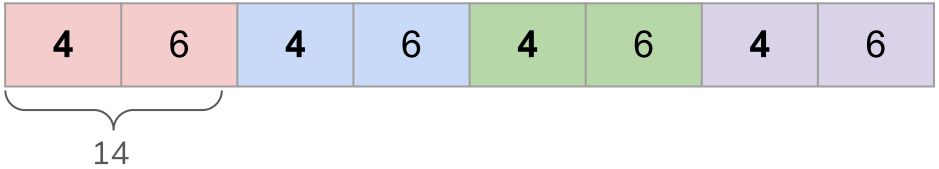
# Subtask 5: A_i + A_(i+1) = 10

- How does this happen?
- Denote k as the number of spares in our previous grouping
- We formed one less spare for a loss of (10+x) points
- For the remaining k-1 spares, we each gain (10-2x) points

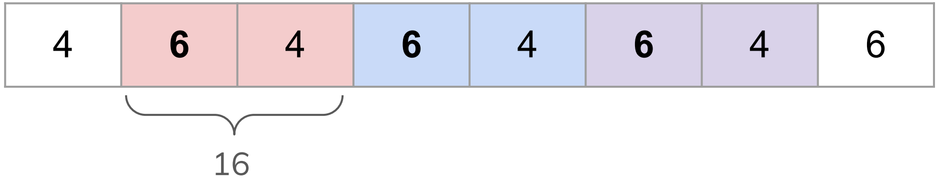

- The two open frames will also give a total of 10 points
- Therefore it is optimal to use the new grouping if (10-2x)*(k-1) > x

## Subtask 5: A_i + A_(i+1) = 10

- Let's look at the counterexample again:




- In the previous grouping, a spare gives 10+4 = 14 points




- If we switch to the new grouping, each spare gains 16-14 = 2 more points
- The open frames gains us 10 points for a total of 16, which is larger than the loss of 14 for forming one less spare

# Subtask 5: A_i + A_(i+1) = 10

- How about the odd N case?
- Consider the following example, and both groupings from before:

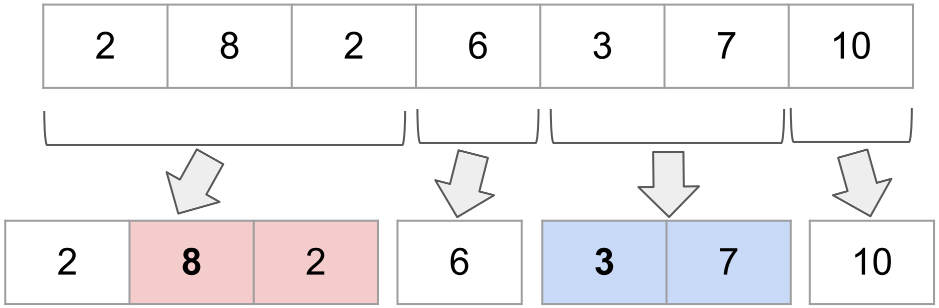
| | | | | | | |
|---|---|---|---|---|---|---|
| **3** | 7 | **3** | 7 | **3** | 7 | 3 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 3 | **7** | 3 | **7** | 3 | **7** | 3 |

- In this example, the first grouping scores a total of (10+3)*3+3 = 42 points
- The second grouping scores a total of 3+(10+7)*3 = 54 points
- Hence we should try both groupings and take the maximum as answer
    - There are no other “more optimal” solutions since we have tried both x and 10-x as the first value of the spare

# Subtask 6: No additional constraints

- From subtask 5, we have a way of handling a “block” of entries that, for some 1 ≤ x ≤ 9, alternates between x and 10-x
- We can split the array into these “blocks”and process them individually
    - For each pair of adjacent elements, they belong to the same block if and only if their sum is 10




- For strikes, we handle them like that of subtask 3