# HW3

## 資工二 宋子維 B03902042

# Problem 1

- ## (1)

Fishy needs to clean the fish at time $t_x$ where $t_i - s \leq t_x \leq t_i - f$ such that *this fish is the first one to be cleaned and satisfies* $t_x + 1 \leq t_i \leq t_x + s$ *which means it is still on the stand at* $t_i$. Thus, Fishy must clean $f - 1$ fish additionally between the time Fishy clean the fish mentioned above and the time $t_i - 1$ so that there are $f$ on the stand at $t_i$. Thus, if $s < f$, it is impossible to satisfy the condition. And if $t_1 - f < 1$, Fishy cannot clean the fish mentioned above at $t_1 - f$.
Therefore, Fishy will be sad with the condition $s < f$ or $t_1 - f < 1$

- ## (2) Dicuss with B03502040

The strategy is to clean the first fish at $t_1 - 1$ to satisfy the condition at $t_1$. If $i$ is the smallest number such that $t_1 + s < t_i$, then Fishy cleans the next fish at $t_i - 1$. If $j$ is the smallest number such that $t_i + s < t_j$, then Fishy cleans the next fish at $t_j - 1$, and so on.

```
// sum is the minimum fiah Fishy needs to clean
// t[] is Fishy's list where t[i] denotes t_i for 1 <= i <= n

sum = 0
last = -s

For i = 1 to n
    if last + s < t[i]
        last = t[i] - 1
        sum += 1
    end if
end For

return sum
```

Let $T_k$ be the optimal solution that Fishy needs to clean the minimum fish to cover $k$ time stamps. $t_i - 1$ is the latest time to clean the fish to satisfy the condition at $t_i$ and $t_i$ is the first one of $k$ time stamps.

### Prove that the problem has optimal substructure.

There exists $t_x \in T_k$ such that the fish cleaned at $t_x$ can be on the stand at $t_i$. Suppose this fish can totally cover $m$ time stamps, then I claim that $|T_k| = 1 + |T_{k-m}|$ and $T_{k-m}$ is the optimal solution to the sub problem. Suppose not, then there exists some $T'_{k-m}$ such that $|T'_{k-m}| < T_{k-m}$. Hence, we can replace $T_{k-m}$ with $T'_{k-m}$, and then we can get $|T'_k| = 1 + |T'_{k-m}| < 1 + |T_{k-m}| = |T_k|$. Then it is a contradiction since

$T_k$ is not the optimal solution. Therefore, the problem has the optimal substructure.

## Prove that the algorithm has greedy choice.

Assume there exists $t_x \in T_k$ with $t_x \leq t_i - 1$ such that the fish cleaned at $t_x$ can be on the stand during $m$ time stamps, that is, $t_x + 1 \leq t_i < t_{i+1} \ldots < t_{i+m-1} \leq t_x + s$ . Since $t_x \leq t_i - 1$, then $t_x + 1 < t_i \leq t_i < t_{i+1} \ldots < t_{i+m-1} \leq t_x + s \leq t_i + s$ which means $t_i - 1$ can cover $p \geq m$ time stamps. Since $T_k$ is the optimal solution, then $|T_k| = 1 + |T_{k-m}|$. If $p > m$, $|T_k| = 1 + |T_{k-m}| > 1 + |T_{k-p}|$ which means $T_k$ is not the optimal solution. Thus, $p = m$. Then we can replace $t_x \in T_k$ with $t_i - 1$, which means $t_i - 1$ is a greeady choice.

- # (3) Dicuss with B03902048

The strategy is the similar to previous one. Howerver, we need to record the fish on the stand at previous time stamp, so we must update information after going through a time stamp.

```
// sum is the minimum fiah Fishy needs to clean
// t[] is Fishy's list where t[i] denotes t_i for 1 <= i <= n
// last[] is used to record when the fish is cleaned

  sum = 0
  For i = 1 to f
      last[i] = -s
  end For

  For i = 1 to n
      index = 1
      For j = 1 to f
          if last[j] + s < t[i]
              last[j] = t[i] - index // update last[j]
              sum += 1
              index += 1
          end if
      end For
  end For

  return sum
```

Let $T_k$ be the optimal solution that Fishy needs to clean the minimum fish to cover $k$ time stamps which are $t_i, t_{i+1} \ldots t_{i+k-1}$ . Each of $t_j, i \leq j \leq i + k - 1$ still needs $w_j$ fish to satisfy the condition. $t_i - 1, t_i - 2, \ldots t_i - w_j$ are the latest time to clean the fish to satisfy the condition at $i$ .

## Prove that the problem has optimal substructure.

Take the same strategy as above. There exists $c_1, c_2, \ldots c_{w_i - 1} \in T_k$ such that the fish cleaned at these time can be on the stand at $t_i$ . Suppose these fish can totally satisfy the condition at the first $m$ time stamps, then I claim that $|T_k| = w_i + |T_{k-m}|$ and $T_{k-m}$ is the optimal solution to the subproblem. Suppose not, then there exists some $T'_{k-m}$ such that $|T'_{k-m}| < T_{k-m}$ . Hence, we can replace $T_{k-m}$ with $T'_{k-m}$ , and then we can get $|T'_k| = w_i + |T'_{k-m}| < w_i + |T_{k-m}| = |T_k|$ . Then it is a contradiction since $T_k$ is not the optimal solution. Therefore, the problem has the optimal substructure.

## Prove that the algorithm has greedy choice.

Assume there exists $s_j \in T_k$ with $s_j \leq t_i - j$ for $j = 1$ to $w_i$ such that these $w_i$ fish cleaned at these time can totally satisfy the condition at the first $m$ time stamps. Same as the previous problem, $t_i - 1, t_i - 2, \ldots t_i - w_j$ can total satisfy the condition at the first $p \geq m$ time stamps. If $p > m$, $|T_k| = w_i + |T_{k-m}| > w_i + |T_{k-p}|$ which means $T_k$ is not the optimal solution. Thus, $p = m$. Then we can replace $s_j \in T_k$ with $t_i - j$ for $j = 1$ to $w_i$, which means $\{t_i - 1, t_i - 2 \ldots t_i - w_i\}$ is a greeady choice.

# Problem 2

- ## (1)

Let $T = \sum_{i=1}^{N} p_i$. Average influence is

$$\frac{b_1(T - p_1 + 1) + b_2(T - p_1 - p_2 + 1) \cdots + b_N}{T}$$
$$= \frac{b_1(p_2 + p_3 \ldots p_N) + b_2(p_3 + p_4 \ldots p_N) + \cdots + b_{n-1}p_N + \sum_{i=1}^{N} b_i}{T}$$
$$= \frac{\sum_{i=1}^{N-1}(b_i(\sum_{j=i+1}^{N} p_i)) + \sum_{i=1}^{N} b_i}{T}$$

- ## (2)

From the terms in the equation above, the only we need to do is change two terms.

$$b_i(T - p_1 - p_2 - p_3 \cdots - p_{i-1} - p_i + 1) + b_{i+1}(T - p_1 - p_2 - p_3 \cdots - p_i - p_{i+1} + 1)$$
$$\downarrow$$
$$b_{i+1}(T - p_1 - p_2 - p_3 \cdots - p_{i-1} - p_{i+1} + 1) + b_i(T - p_1 - p_2 - p_3 \cdots - p_i - p_{i+1} + 1)$$

Hence the change in the numerator is

$$\begin{aligned}
&b_{i+1}(T - p_1 - p_2 - p_3 \cdots - p_{i-1} - p_{i+1} + 1) + b_i(T - p_1 - p_2 - p_3 \cdots - p_i - p_{i+1} + 1) \\
&- b_i(T - p_1 - p_2 - p_3 \cdots - p_{i-1} - p_i + 1) - b_{i+1}(T - p_1 - p_2 - p_3 \cdots - p_i - p_{i+1} + 1) \\
&= -b_i p_{i+1} + b_{i+1} p_i
\end{aligned}$$

Then the total change is

$$\frac{-b_i p_{i+1} + b_{i+1} p_i}{T}$$

- ## (3)

$$p \rightarrow q \Leftrightarrow !q \rightarrow !p$$

To prove this, it is equal to show if A is not an optimal solution, then for some $i < j$, there exists some $S$ s.t $S_k = A_i, S_{k+1} = A_j$ and $f(S_{swap(k,k+1)}) > f(S)$.
Suppose $A$ is not an optimal solution, then there exists some $A'$ with different permutation with $A$ such that

$f(A') > f(A)$. Since there are totally $N! < \infty$ permutations, we can always get $A'$ via a finite series of swapping two adjacent elements. Suppose it needs to perform $p$ swaps to get $A'$ and each change of the average influence is $c_i$, $1 \le i \le p$. Then we can get the relation

$$f(A') = f(A) + \sum_{i=1}^{p} c_i$$

And since $f(A') > f(A)$, it means $\sum_{i=1}^{p} c_i > 0$. Hence, there exists $c_q > 0$ for some $1 \le q \le p$. Let $S$ be the sequence after swapping $q-1$ times. Since we always swap two adjacent elements, there exists $i < j$ such that $S_k = A_i$, $S_{k+1} = A_j$ clearly. By the previous problem, because $c_q = \frac{-b_i p_j + b_j p_i}{T} = f(S_{swap(k,k+1)}) - f(S) > 0$, $f(S_{swap(k,k+1)}) > f(S)$, Q.E.D.

- ## (4)

By problem (3), our goal is to find a sequence $A$ and there does not exist $S$ such that $S_k = A_i$, $S_{k+1} = A_j$ and $f(S_{swap(k,k+1)}) > f(S)$ for some $i < j$. By problem (2), we know

$$\begin{aligned} & f(S_{swap(k,k+1)}) > f(S) \\ \to\ & f(S_{swap(k,k+1)}) - f(S) > 0 \\ =\ & \frac{-b_i p_j + b_j p_i}{T} > 0 \\ \to\ & b_j p_i > b_i p_j \\ \to\ & \frac{b_j}{p_j} > \frac{b_i}{p_i} \end{aligned}$$

Hence, we can calculate $\frac{influence}{price} = \frac{b_i}{p_i}$ of each equipment $e_i$ which takes $O(n)$. Then sort $e_i$ in decreasing order of $\frac{b_i}{p_i}$. It takes $O(n \log n)$ if using quick sort. Sequence $A$ with ordered {$e$} is the optimal solution and the complexity is $O(n) + O(n \log n) = O(n \log n)$.

```
// A is an origin sequence with unordered {e}
// b[i] and p[i] denotes b_i and p_i respectively for 1 <= i <= n
// w[i] is used to store b[i] / p[i]

For i = 1 to n
    w[i] = b[i] / p[i] // calculate b_i / p_i

quick_sort(A, w) // use quick sort to sort A according to b_i / p_i

// A is the optimal solution with maximum average influence
```

## Prove the correctness

Since $A$ is a sequence of ordered {$e$}, then $w_1 \ge w_2 \cdots \ge w_n$, where $w_i = \frac{b_i}{p_i}$. Thus, there does not exist $i < j$ such that $\frac{b_j}{p_j} = w_j > w_i = \frac{b_i}{p_i}$. By the deduction above, $A$ is the optimal solution.

- ## (5)

Since float point calculation is forbidden, $\frac{b_i}{p_i}$ can not be calculated directly. Hence, to compare $\frac{b_i}{p_i}$ to $\frac{b_j}{p_j}$, we can cast them and perform cross-multiplication on it. That is, if $\frac{b_i}{p_i} > \frac{b_j}{p_j}$, then $(long\,long)(b_i) * (p_j) > (long\,long)(b_j) * (p_i)$. Just modify the comparison method in the previous problem.

# Problem 3

- **(1)**

## Prove that Paul has a strategy to get at least a score of $\sum_{j\ is\ odd} a_{[j]1}$

Paul has a strategy that he always takes the card with the maximum number to get at least a score of $\sum_{j\ is\ odd} a_{[j]1}$. Suppose Paul takes $\{b_1, b_2 \ldots b_{\lceil \frac{N}{2} \rceil}\}$ with $b_1 > b_2 \cdots > b_{\lceil \frac{N}{2} \rceil}$. I claim that

$$\forall i\ ,\ b_i \geq a_{[2i-1]1}$$

Clearly, when $i = 1, b_1 = a_{[1]1}$ Let the number of remaining cards in range $[a_{[1]1}, a_{[j]1}]$ be $w_j$. After Paul takes $b_k$, Paul and John totally take $2k - 1$ cards. Thus $w_{2k+1} \geq 2$. John takes one card arbitrarily in his turn, so $w_{2k+1}$ "may" minus 1, that is, $w_{2k+1} \geq 1$. When Paul takes $b_{k+1}$, since Paul always takes the maximum one of remaining card and $w_{2k+1} \geq 1$, then $b_{k+1} \geq a_{[2k+1][1]}$. Since $\forall i\ ,\ b_i \geq a_{[2i-1]1}$, then

$$\sum_{i=1}^{\lceil \frac{N}{2} \rceil} b_i \geq \sum_{j\ is\ odd} a_{[j]1}$$

## Prove that John has a strategy to get at least a score of $\sum_{j\ is\ even} a_{[j]1}$

John has a strategy that he always takes the card with the maximum number to get at least a score of $\sum_{j\ is\ odd} a_{[j]1}$. Suppose Paul takes $\{c_1, c_2 \ldots c_{\lfloor \frac{N}{2} \rfloor}\}$ with $c_1 > c_2 \cdots > c_{\lfloor \frac{N}{2} \rfloor}$. I claim that

$$\forall i\ ,\ c_i \geq a_{[2i]1}$$

When $i = 1, c_1 = a_{[1]1} \geq a_{[2i]1}$ if Paul doesn't take $a_{[1]1}$ as his first card or $c_1 = a_{[2]1} \geq a_{[2i]1}$ if Paul takes $a_{[1]1}$ as his first card. Let the number of remaining cards in range $[a_{[1]1}, a_{[j]1}]$ be $w_j$. After John takes $c_k$, Paul and John totally take $2k$ cards. Thus $w_{2k+2} \geq 2$. Paul takes one card arbitrarily in his turn, so $w_{2k+2}$ "may" minus 1, that is, $w_{2k+2} \geq 1$. When John takes $c_{k+1}$, since John always takes the maximum one of remaining card and $w_{2k+2} \geq 1$, then $c_{k+1} \geq a_{[2k+2][1]}$. Since $\forall i\ ,\ c_i \geq a_{[2i]1}$, then

$$\sum_{i=1}^{\lfloor \frac{N}{2} \rfloor} c_i \geq \sum_{j\ is\ even} a_{[j]1}$$

- **(2)**

## Prove that Paul has a strategy to get at least a score of

$\sum_{i=1}^{N} \sum_{j=1}^{n_i/2} a_{ij}$

Paul's strategy is to take the card from the same pile where John takes the card in the previous turn. If there is not any card in that pile or it is Paul's first turn, Paul takes the card from an arbitrary pile.

Since each $n_i$ is even, both Paul and John takes $X$ cards. Let $P_{[i][j]}$ ($H_{[i][j]}$) be the number of cards which Paul(John) takes from pile $j$ after Paul's $i-th$ turn but before John's $i-th$ turn. I claim that for $1 \le i \le X$ there exists one and only one $m$ such that $P_{[i][m]} - H_{[i][m]} = 1$ and $P_{[i][j]} = H_{[i][j]}$ for other $j \neq m$. I will show it by induction.

When $i = 1$, it holds clearly. Suppose it holds when $i = k$. That is, there exists one and only one $m$ such that $P_{[k][m]} - H_{[k][m]} = 1$ and $P_{[k][j]} = H_{[k][j]}$ for other $j \neq m$. Then there are four cases when $i = k + 1$.
Case 1: John takes the card from pile $m$, and pile $m$ is still not empty.
Since pile $m$ is still not empty, then Paul needs to follow John to take the card from pile $m$. Hence,
$P_{[k+1][m]} - H_{[k+1][m]} = (P_{[k][m]} + 1) - (H_{[k][m]} + 1) = 1$ and $P_{[k][j]} = H_{[k][j]}$ for other $j \neq m$.

Case 2: John takes the card from pile $m$, and pile $m$ is empty after John's turn.
Since pile $m$ is empty after John's turn, Paul need to take the card from a non-empty pile $m'$. Hence,
$P_{[k+1][m]} - H_{[k+1][m]} = P_{[k][m]} - (H_{[k][m]} + 1) = 0$ but
$P_{[k+1][m']} - H_{[k+1][m']} = (P_{[k][m']} + 1) - H_{[k][m']} = 1$ and
$P_{[k+1][j]} - H_{[k+1][j]} = P_{[k][j]} - H_{[k][j]} = 0$ for other $j \neq m'$.

Case 3: John takes the card from pile $l \neq m$, and the pile $l$ is still not empty.
Since pile $l$ is still not empty, then Paul needs to follow John to take the card from pile $l$.
Hence, $P_{[k+1][m]} - H_{[k+1][m]} = P_{[k][m]} - H_{[k][m]} = 1$,
$P_{[k+1][l]} - H_{[k+1][l]} = (P_{[k][l]} + 1) - (H_{[k][l]} + 1) = 0$ and
$P_{[k+1][j]} - H_{[k+1][j]} = P_{[k][j]} - H_{[k][j]} = 0$ for other $j \neq m$.

Case 4: John takes the card from pile $l \neq m$, and pile $l$ is empty after John's turn.
This situation doesn't exist since $n_l$ is even. Since $P_{[k][l]} = H_{[k][l]} = R$ and pile $l$ is empty after John's turn, $n_l = 2R + 1$ is odd. Hence, it is a contraction.

Therefore, it holds for all $1 \le i \le X$ by induction. When $i = X$, there exists one and only one $m$ such that $P_{[X][m]} - H_{[X][m]} = 1$. Since each pile has even number of cards, the last card John need to take is in pile $m$. Thus, when the game ends, Paul and John take the same number of cards in the same pile. Since Paul always takes the cards from the top, Paul takes $a_{i1}$ to $a_{i(\frac{n_i}{2})}$, $1 \le i \le N$. Therefore Paul can get at least $\sum_{i=1}^{N} \sum_{j=1}^{n_i/2} a_{ij}$.

## Prove that Paul has a strategy to get at least a score of $\sum_{i=1}^{N} \sum_{j=n_i/2+1}^{n_i} a_{ij}$

John's strategy is to take the card from the same pile where Paul takes the card in the previous turn.

Since each $n_i$ is even, both Paul and John takes $X$ cards. Let $P_{[i][j]}$ ($H_{[i][j]}$) be the number of cards which Paul(John) takes from pile $j$ after John's $i-th$ turn. I claim that $P_{[i][j]} = H_{[i][j]}$ for each $i$ and $j$. Proof is also by induction.

When $i = 1$, it holds clearly. Suppose it holds when $i = k$. That is, $P_{[k][j]} = H_{[k][j]}$ for each $j$. Then there are two cases when $i = k + 1$.

Case 1: Paul arbitrarily takes the card from a non-empty pile $m$, and the pile is empty after Paul's turn.

This situation doesn't exist since $n_m$ is even. Since $P_{[k][m]} = H_{[k][m]} = R$ and pile $l$ is empty after John's turn, $n_m = 2R + 1$ is odd. Hence, it is a contraction.

Case 2: Paul arbitrarily takes the card from a non-empty pile $m$, and the pile is still not empty after Paul's turn. Since $P_{[k][m]} = H_{[k][m]}$, then $P_{[k+1][m]} = P_{[k][m]} + 1 = H_{[k][m]} + 1 = H_{[k+1][m]} + 1$. And other terms are the same as $i = k$.

Therefore, it holds for all $1 \le i \le X$ by induction. Thus, when the game ends, Paul and John take the same number of cards in the same pile. Since John always takes the cards from the botom, John takes $a_{i1}$ to $a_{i(\frac{n_i}{2})}$, $1 \le i \le N$. Therefore, John can get at least $\sum_{i=1}^{N} \sum_{j=n_i/2+1}^{n_i} a_{ij}$.

- **(3)**

## Prove that Paul has a strategy to get at least a score of $\sum_{i=1}^{N} \sum_{j=1}^{\frac{n_i-1}{2}} a_{ij} + \sum_{k\ is\ odd} a_{[k]\frac{n_i+1}{2}}$

In Paul's first turn, Paul takes the card from the top of the pile with $a_{[1]\frac{n_i+1}{2}}$. In other Paul's turns, Paul takes the card from the same pile where John takes the card in the previous turn if this pile is not empty after John's turn. If it is empty, then Paul takes the card from the pile with maximum $a_{[k]\frac{n_i+1}{2}}$ among the piles that Paul and John take the same number of cards from them.

Let $P_j(H_j)$ be the number of cards that Paul(John) take from pile $i$ after Paul's $j-th$ turn but before John's $j-th$ turn for each $i$. I claim that if there exists a $s$ such that $P_s = H_s + 1$ for pile $i$, then for all $j \ge s$, $P_j = H_j + 1$ for pile $i$.

When $j = s$, $P_s - H_s = 1$ holds clearly. Suppose it holds when $j = m$, that is, $P_m - H_m = 1$. Then there are four cases when $j = m + 1$.

Case 1: John takes a card from pile $i$ in his $m - th$ turn, and pile $i$ is still not empty after John's turn.
Since pile $i$ is still not empty, Paul needs to follow John to take a card from pile $i$.
Then $P_{m+1} - H_{m+1} = (P_m + 1) - (H_m + 1) = 1$.

Case 2: John takes a card from pile $i$ in his $m - th$ turn, and pile $i$ is empty after John's turn.
The situation doesn't exist since $n_i$ is odd. Since $P_m = H_m + 1$ and pile $l$ is empty after John's turn, $n_i = P_m + H_m + 1 = 2H_m + 2$ is even. Hence, it is a contraction.

Case 3: John takes a card from pile $l \ne i$ in his $m - th$ turn, and pile $l$ is still not empty after John's turn.
$P_{m+1} - H_{m+1} = P_m - H_m = 1$

Case 4: John takes a card from pile $l \ne i$ in his $m - th$ turn, and pile $l$ is after John's turn.
Since pile $l$ is empty, Paul needs to take the card from the pile which has maximum $a_{[k]\frac{n_i+1}{2}}$ among the piles that Paul and John take the same number of cards from them. However, $P_m = H_m + 1 \ne H_m$, so Paul will not take a card from pile $i$. Thus, $P_{m+1} - H_{m+1} = P_m - H_m = 1$.

Hence, if there exists if there exists a $s$ such that $P_s = H_s + 1$, then for all $j \ge s$, $P_j = H_j + 1$ by induction. If $P_j = H_j + 1$ when the pile $i$ is empty, we can get $P_j = \frac{n_i+1}{2}$ by solving

$$\begin{cases} P_j = H_j + 1 \\ P_j + H_j = n_i \end{cases}$$

That is, once Paul takes one more card from the same pile $i$ than John and since Paul takes from the top, Paul can get $\{a_{i1}, a_{i2}, \ldots a_{i\frac{n_i+1}{2}}\}$.

For the pile $i$ which doesn't exist a $s$ such that $P_s = H_s + 1$, I claim that $H_j = P_j$ for all $j$. The proof is similar to the previous one. Then we can deduce that John can takes one more card than Paul when pile $i$ is empty by this fact. That is, Paul can get

$$\{a_{i1}, a_{i2}, \ldots a_{i\frac{n_i-1}{2}}\}$$

Since Paul first take the card from the pile $p$ with $a_{[1]\frac{n_p+1}{2}}$, then $P_1 = 1 = 0 + 1 = H_1 + 1$. Thus Paul can get

$$\{a_{p1}, a_{p2}, \ldots a_{p\frac{n_p-1}{2}}, a_{p\frac{n_p+1}{2}} = a_{[1]\frac{n_p+1}{2}}\}$$

And since Paul takes the card from the pile with maximum $a_{[k]\frac{n_i+1}{2}}$ among the piles that Paul and John take the same number of cards from them if the pile which John takes the card from is empty, it means that once John takes some $a_{[m]\frac{n_i+1}{2}}$, Paul can have another $a_{[k]\frac{n_i+1}{2}}$ since $P_s = H_s + 1$ after Paul takes the card from the pile with $a_{[k]\frac{n_i+1}{2}}$. It implies that *Paul and John take turns to get* $a_{[k]\frac{n_i+1}{2}}$ *and Paul takes* $a_{[1]\frac{n_i+1}{2}}$ *first*. Thus, this problem is now the same as problem (1), and then we can deduce Paul can get at least a score of

$$\sum_{i=1}^{N} \sum_{j=1}^{\frac{n_i-1}{2}} a_{ij} + \sum_{k \text{ is odd}} a_{[k]\frac{n_i+1}{2}}$$

by the result of problem (1) and the deduction of Paul's strategy.

## Prove that Paul has a strategy to get at least a score of $\sum_{i=1}^{N} \sum_{j=(n_i+3)/2}^{n_i} a_{ij} + \sum_{k \text{ is even}} a_{[k]\frac{n_i+1}{2}}$

John has a strategy that John takes the card from the pile which Paul takes the card from in the previous turn. If this pile is empty, then John takes the card from the pile with maximum $a_{[k]\frac{n_i+1}{2}}$ among the piles that Paul and John take the same number of cards from them.

Let $P_j(H_j)$ be the number of cards that Paul(John) take from pile $i$ after John's $j-th$ turn for each $i$. Then we can use the same method in the previous problem to prove that if there exists a $s$ such that $H_s = P_s + 1$ for pile $i$, then for all $j \geq s$, $H_j = P_j + 1$ for pile $i$. One can reduce that John can get

$$\{a_{in_i}, a_{i(n_i-1)}, \ldots a_{i\frac{n_i+1}{2}}\}$$

For other pile $p$ which doesn't exist a $s$ such that $H_s = P_s + 1$, we can apply the same method in the previous problem to deduce that John can get

$$\{a_{pn_p}, a_{p(n_p-1)}, \ldots a_{p\frac{n_p+3}{2}}\}$$

And since John takes the card from the pile with maximum $a_{[k]\frac{n_i+1}{2}}$ among the piles that Paul and John take the same number of cards from them if the pile which Paul takes the card from is empty, it means that once Paul takes some $a_{[m]\frac{n_i+1}{2}}$, John can have another $a_{[k]\frac{n_i+1}{2}}$ since $P_s = H_s + 1$ after John takes the card from the pile with $a_{[k]\frac{n_i+1}{2}}$ . It implies that *Paul and John take turns to get* $a_{[k]\frac{n_i+1}{2}}$ *and Paul takes the first turn*. Thus, this problem is now the same as problem (1), and then we can deduce Paul can get at least a score of

$$\sum_{i=1}^{N} \sum_{j=(n_i+3)/2}^{n_i} a_{ij} + \sum_{k\ is\ even} a_{[k]\frac{n_i+1}{2}}$$

by the result of problem (1) and the deduction of John's strategy.

- ## (4)

If both Paul and John play optimally, then Paul will get

$$\sum_{n_i\ is\ even} \sum_{j=1}^{\frac{n_i}{2}} a_{ij} + \sum_{n_i\ is\ odd} \sum_{j=1}^{\frac{n_i-1}{2}} a_{ij} + \sum_{k\ is\ odd} a_{[k]\frac{n_i+1}{2}}$$

and John will get

$$\sum_{n_i\ is\ even} \sum_{j=n_i/2+1}^{n_i} a_{ij} + \sum_{n_i\ is\ odd} \sum_{j=(n_i+3)/2}^{n_i} a_{ij} + \sum_{k\ is\ even} a_{[k]\frac{n_i+1}{2}}$$

Note that $a_{[k]\frac{n_i+1}{2}}$ denotes the $k$-th largest $a_{i\frac{n_i+1}{2}}$ where $n_i$ is odd.

Let the score that Paul(John) gets be $S_{Paul}(S_{John})$.
From the previous two problems, we know that Paul will take his first card with $a_{[1]\frac{n_i+1}{2}}$ . Then since John plays optimally, he follows Paul to take the card from the same pile if this pile is not empty. If this pile is empty, John takes the card from the pile with $a_{[2]\frac{n_i+1}{2}}$ . Next, Paul follows John, and so on. Because of their strategies, the piles with odd $n_i$ will be empty first. Hence, they start to take the card with even $n_i$ and follow their strategies. The relation of $S_{Paul}$ and $S_{John}$ can be written as

$$S_{Paul} \geq \sum_{n_i\ is\ even} \sum_{j=1}^{\frac{n_i}{2}} a_{ij} + \sum_{n_i\ is\ odd} \sum_{j=1}^{\frac{n_i-1}{2}} a_{ij} + \sum_{k\ is\ odd} a_{[k]\frac{n_i+1}{2}} = A$$
$$S_{John} \geq \sum_{n_i\ is\ even} \sum_{j=n_i/2+1}^{n_i} a_{ij} + \sum_{n_i\ is\ odd} \sum_{j=(n_i+3)/2}^{n_i} a_{ij} + \sum_{k\ is\ even} a_{[k]\frac{n_i+1}{2}} = B$$

by the previous problems clearly.

Then we can deduce

$$S_{Paul} = \sum_{n_i\ is\ even} \sum_{j=1}^{\frac{n_i}{2}} a_{ij} + \sum_{n_i\ is\ odd} \sum_{j=1}^{\frac{n_i-1}{2}} a_{ij} + \sum_{k\ is\ odd} a_{[k]\frac{n_i+1}{2}}$$
$$S_{John} = \sum_{n_i\ is\ even} \sum_{j=n_i/2+1}^{n_i} a_{ij} + \sum_{n_i\ is\ odd} \sum_{j=(n_i+3)/2}^{n_i} a_{ij} + \sum_{k\ is\ even} a_{[k]\frac{n_i+1}{2}}$$

by hint since

$$A + B = \sum_{i=1}^{N} \sum_{j=1}^{n_i} a_{ij} = S_{Paul} + S_{John}$$