**2. Розбір задачі SECTION**

Нехай π – перестановка порядку N. За умовою задачі, загальна ефективність роботи гуртка не змінюватиметься при довільному розбитті на пари, якщо для всіх перестановок π виконуватиметься наступна рівність:

A[1, π1] + A[2, π2] + A[3, π3] + … + A[N, πN] = const. (1)

Розглянемо гурток з (K+1) досвідченими програмістами та (K+1) новачками. Припустимо, що для перших K досвідчених програмістів та перших K новачків умова (1) виконується (тобто загальна ефективність гуртка, утвореного з цих учнів, є сталою) і результат їх роботи рівний S. Для (К+1)-го досвідченого програміста та (К+1)-го новачка можливі два принципово різні варіанти вибору пари:

1. (К+1)-ий досвідчений програміст працюватиме з (K+1)-им новачком (ефективність роботи гуртка буде рівна (S + A[K+1, K+1]);
2. (К+1)-ий досвідчений програміст працюватиме з πi – им новачком, а i-ий – з (K+1)-им новачком.

Незалежно від того, як будуть обрані пари для (К+1)-го досвідченого програміста та (К+1)-го новачка, ефективність роботи гуртка повинна бути рівна значенню (S + A[K+1, K+1]). Для довільного розбиття на пари, має виконуватися умова:

A[1, π1] + A[2, π2] + … + A[i, πi] + … + A[K, πK] + A[K+1, K+1] =

= A[1, π1] + A[2, π2] + … + A[i-1, πi-1] + A[i, K+1] + A[i+1, πi+1] +… + A[K, πK] + A[K+1, πi],

виконавши нескладні арифметичні операції, маємо:

A[i, πi] + A[K+1, K+1] = A[i, K+1] + A[K+1, πi].

Зважаючи на те, які значення можуть набирати значення i та πi, останню умову можна записати наступним чином:

A[i, j] + A[K+1, K+1] = A[i, K+1] + A[K+1, j],

для всіх 1 ≤ i, j ≤ K.

Вищеописана умова дозволяє перевірити гурток з (К+1) досвідченого програміста та (К+1) новачка, враховуючи те, що ефективність гуртка утвореного з К досвідчених програмістів та К новачків є сталою. Оскільки для меншої кількості учнів повинна виконуватися аналогічна умова, то можна записати загальну умову для N досвідчених програмістів та N новачків:

A[i1, j1] + A[i2, j2] = A[i1, j2] + A[i2, j1], (3)

для всіх 1 ≤ i1, j1, i2, j2 ≤ N.

Для перевірки цих умов необхідно O(N4) операцій.

Базою такої індукції буде гурток з двох досвідчених програмістів та двох новачків, для нього необхідною і достатньою, щоб виконувалася умова A[1, 1] + A[2, 2] = A[1, 2] + A[2, 1].

Спробуємо дещо модифікувати умову (3). Перепишемо її наступним чином:

A[i1, j] + A[i2, j + x] = A[i1, j + x] + A[i2, j].

Припустимо, що для всіх y < x попередня умова виконується, тоді маємо:

A[i1, j] + A[i2, j + y] = A[i1, j + y] + A[i2, j]

A[i1, j+ y] + A[i2, j + x] = A[i1, j + x] + A[i2, j+ y].

Додавши дві попередні рівності, маємо:

A[i1, j] + A[i2, j + x] = A[i1, j + x] + A[i2, j],

тобто доведення потрібної розмірності базується на попередньому доведенні менших розмірностей, зменшуючи як по першій координаті, так і по другій. Тепер стає очевидно, щоб довести умову (3), достатньо перевірити наступні умови:

A[i, j] + A[i+1, j+1] = A[i+1, j] + A[i, j+1],

для всіх 1 ≤ i, j < N.

Застосувавши вказаний прийом, розв’язання задачі потребуватиме O(N2) операцій.