# 아름다운 수열 - 알고리즘 문제 요약

## 문제

길이 N짜리 수열 A에서 길이 K 이상의 연속 부분 수열을 하나 골라 아래의 "아름다운 정도"를 최대화하라.

아름다운 정도 = (A₁² + A₂² + ... + Aₙ²) / (A₁ + A₂ + ... + Aₙ)

모든 수는 정수이며, 수열의 각 원소는 1 이상 10^6 이하의 자연수이다. 답은 실수로 출력하며, 절대 또는 상대 오차가 1e-6 이하인 경우 정답으로 인정된다.

## 예시

|  |
| --- |
| 예제 입력 1: 3 1 10 8 14  출력: 14.0000000000  예제 입력 2: 3 2 2 1 2  출력: 1.8000000000 |

## 해법

1. 목표는 어떤 연속된 구간의 "제곱의 합 / 합"을 최대화하는 것.

2. 이분 탐색을 통해 아름다운 정도 x를 고정하고, 이를 만족하는 구간이 존재하는지 확인한다.

- A[i]^2 - x\*A[i] 를 B[i]로 두고, B[i]의 누적합에서 길이 K 이상 구간의 합이 0 이상인 것이 존재하는지 판별.

3. 누적합 배열 P[i]를 만들고, 각 위치 i에서 P[i] - min(P[0..i-K]) 가 0 이상인 경우가 존재하면 된다.

4. 이를 통해 조건을 만족하는 x의 최댓값을 이분 탐색으로 찾아낸다.

## 복잡도

- 이분 탐색: O(log(범위)) ≒ 60

- 각 판별 단계는 O(N)

- 전체 시간복잡도: O(N log(maxA)) ≒ 6 \* 10^6 (N=10^5에서 충분히 빠름)

## 요약

- 핵심 아이디어: 평균 최대화는 매개변수 탐색으로 바꾸고, B[i] = A[i]^2 - x\*A[i] 를 구성하여 길이 K 이상 부분합이 양수인 구간이 있는지 확인한다.

- 핵심 로직: prefix[j] - min(prefix[0..j-K]) ≥ 0 인 j가 존재하는지를 선형 스캔으로 판별.

- 대표 테크닉: 이분 탐색 + 누적합 최솟값 갱신 → O(N log 범위)로 해결.

## C++ 코드

|  |
| --- |
| #include <bits/stdc++.h>  using namespace std;    // 아름다운 정도가 x 이상인 부분 수열이 존재하는지 확인하는 함수  bool ok(const vector<int>& A, int K, double x) {  int N = A.size();  vector<double> prefix(N + 1);  for (int i = 1; i <= N; ++i)  // B[i] = A[i]^2 - x \* A[i]  prefix[i] = prefix[i - 1] + A[i - 1] \* A[i - 1] - x \* A[i - 1];    double min\_pref = 0.0; // 최소 prefix 값 (초기에는 P[0])  for (int j = K; j <= N; ++j) {  min\_pref = min(min\_pref, prefix[j - K]); // j-K까지 중 최소 prefix  if (prefix[j] - min\_pref >= 0.0) // 조건 만족하는 구간 발견  return true;  }  return false;  }    int main() {  ios::sync\_with\_stdio(false);  cin.tie(nullptr);    int N, K;  cin >> N >> K;  vector<int> A(N);  int maxA = 0;  for (int &a : A) {  cin >> a;  maxA = max(maxA, a);  }    // 이분 탐색으로 아름다운 정도의 최댓값 탐색  double lo = 0.0, hi = maxA;  for (int iter = 0; iter < 60; ++iter) {  double mid = (lo + hi) / 2;  if (ok(A, K, mid)) lo = mid;  else hi = mid;  }    cout << fixed << setprecision(10) << lo << '\n';  return 0;  } |

https://codejam.lge.com/contest/problem/1520/3