G - Highest Ratio Editorial by en_translator

Consider $(N+1)$ points numbered $0, 1, 2, \dots, N$ on a two-dimensional plane.
Here, point 0's coordinates are $(x_0, y_0) = (0, 0)$, and point i's are $(x_i, y_i) = (i, A_1 + A_2 + \cdots + A_i)$.
Then, the average value of $A_L, A_{L+1}, \dots, A_R$ is represented as the slope of points $L-1$ and points R.

For a fixed k $(1 \leq k \leq N)$, let us consider how to find the maximum possible average value of the k-th through r-th terms of the sequence A.
Take the convex hull of points $k-1, k, k+1, \dots, N$. If points $k-1, k, k+1, \dots, N$ are colinear, then $A_k = A_{k+1} = \cdots\ A_N$, so the average is A_k independent of A_k.
Otherwise, the convex hull is a region with a positive area. Point $k-1$, which is the only point with the minimum x coordinate, is on the boundary of the convex hull.
When you traverse the boundary clockwise from point $(k-1)$, if the next point is p, then the answer is $\frac{y_p - y_{k-1}}{x_p - x_{k-1}}$. This is because, when you do so **counterclockwise** from point $(k-1)$, if the next point is q, then for a point i $(k \leq i \leq N)$ that forms the convex hull, the slope of the line passing through points i and $(k-1)$ is at least that for points q and points $(k-1)$, and at most that for points p and $(k-1)$, by definition of convex hull. The value we seek for is equivalent to the maximum slope of the line connecting point i $(k \leq i \leq N)$ and point $(k-1)$, so the answer is $\frac{y_p - y_{k-1}}{x_p - x_{k-1}}$, which is the slope of the line connecting points p and $(k-1)$ $(k-1)$ $(k-1)$ $(k-1)$.

After all, for each k, it is sufficient to consider the convex hull of points $k-1, k+1, \dots, N$ and find the next point you encounter when traversing the boundary of the convex hull clockwise from the point $(k-1)$; in other words, the point next to point $(k-1)$ in the point sequence that forms the upper hull.
If you naively find the convex hull for each k, it exceeds the execution time limit. However, when you find the upper convex hull while adding points $N, N-1, \dots, 0$ in this order, you can find the slope of the line passing through point $k-1$ and the point next to it in the point sequence that forms the upper hull. This way, all the desired value can be found in the same time complexity as finding the convex hull of the point set consisting of points $0, 1, \dots, N$ only once.
Since the points are sorted by x coordinates, once the coordinates are found, one can find the answer in a total of $O(N)$ time using algorithms like Graham scan.
Hence, the problem has been solved.

Sample code in C++:

```cpp
#include <bits/stdc++.h>
using namespace std;
int main(void){
        int n;
        long double x,y;
        cin>>n;
        vector<long double>a(n+1,0.0);
        vector<long double>ans(n);
        vector<int>b;
        for(int i=0;i<n;i++){
                cin>>a[i+1];
                a[i+1]+=a[i];
        }
        int bsz=0;
        for(int i=n;i>=0;i--){
                while(b.size()>1){
                        x=(a[b[bsz-1]]-a[i])*((long double)(b[bsz-2]-i));
                        y=(a[b[bsz-2]]-a[i])*((long double)(b[bsz-1]-i));
                        if(x<=y){
                                b.pop_back();
                                bsz--;
                        }
                        else break;
                }
                if(i<n)ans[i]=(a[b[bsz-1]]-a[i])/((long double)(b[bsz-1]-i));
                b.push_back(i);
                bsz++;
        }
        for(int i=0;i<n;i++)cout<< std::fixed << std::setprecision(10)<<ans[i]<<endl;
        return 0;
}
```