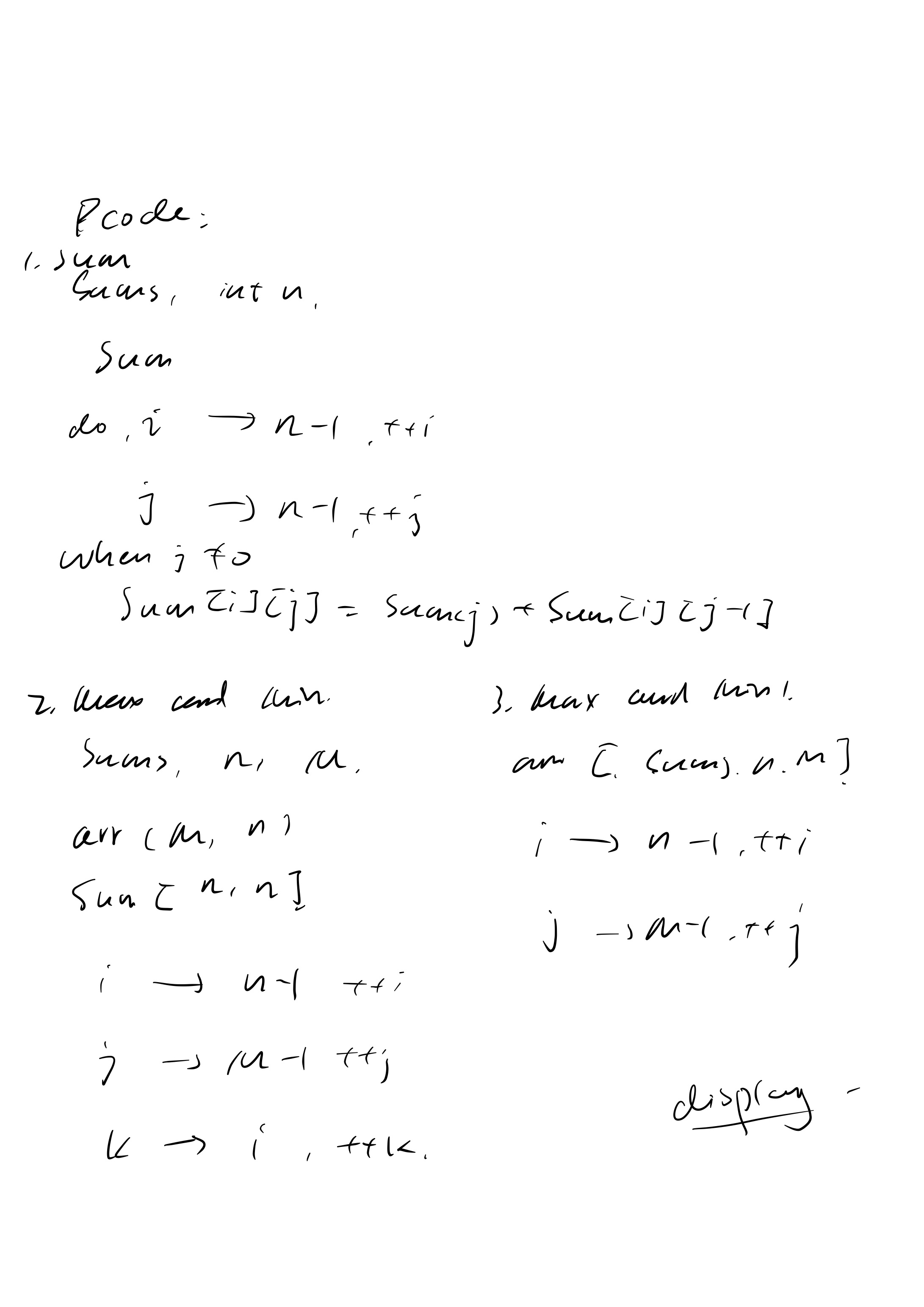
7205 Project 1

JinjingMiao

Pseudocode



T(n)= O(n^2)

Code:

\#include **<iostream>**#include **<vector>**#include **<algorithm>  
  
using namespace** std;  
  
vector<vector<**int**>> SUM(vector<**int**> &sums, **int** n)  
{  
 vector<vector<**int**>> Sum(n, vector<**int**> (n, 0));  
 **int** i;  
 **int** j;  
 **for** ( i = 0; i <= n - 1; ++i)  
 {  
 **for** ( j = i; j <= n - 1; ++j)  
 {  
 **if** (j == 0)  
 {  
 Sum[i][j] = sums[j];  
 }  
 **else** {  
 Sum[i][j] = sums[j] + Sum[i][j - 1];  
 }  
 }  
 }  
 **return** Sum;  
}  
  
  
vector<vector<**int**>> MaxMin(vector<**int**> &sums, **int** n, **int** M)  
{  
 vector<vector<**int**>> arr(M, vector<**int**> (n, 0));  
 vector<vector<**int**>> Sum(n, vector<**int**> (n, 0));  
 Sum = SUM(sums, n);  
 **int** i;  
 **int** j;  
 **int** k;  
 **for** ( i = 0; i <= n - 1; ++i)  
 {  
 arr[0][i] = Sum[0][i];  
 }  
 **for** ( j = 1; j <= M - 1; ++j)  
 {  
 **for** ( i = 0; i <= n - 1; ++i)  
 {  
 **int** max\_num = 0, min\_num = 0;  
 **for** ( k = j - 1; k < i; ++k)  
 {  
 max\_num = max(max\_num, min\_num);  
 min\_num = min(arr[j - 1][k], Sum[k + 1][i]);  
  
 }  
 arr[j][i] = max\_num;  
 }  
 }  
 **return** arr;  
}  
  
  
vector<**int**> MaxMin1(vector<**int**> &sums, **int** n, **int** M)  
{  
 vector<**int**> Group(M);  
 vector<vector<**int**>> arr(M, vector<**int**> (n, 0));  
 arr = MaxMin(sums, n, M);  
 **int** i;  
 **int** j;  
 **for** ( j = 0; j <= M - 1; ++j)  
 {  
 **for** ( i = 0; i <= n - 1;++i)  
 {  
 **if** (arr[j][i] == arr[M - 1][n - 1])  
 {  
 Group[j] = i + 1;  
 }  
 **else if**(arr[j][i] < arr[M - 1][n - 1])  
 {  
 Group[j] = i + 2;  
 }  
  
 }  
  
 }  
 **int** m;  
 **for** ( m = M - 1; m >= 1; --m)  
 {  
 Group[m] = Group[m] - Group[m - 1];  
 }  
 **return** Group;  
}  
  
**int** main()  
{  
 vector<**int**> sums = {3, 9, 7, 8, 2, 6, 5, 10, 1, 7, 6, 4};  
 **int** n = sums.size();  
 **int** M = 3;  
  
  
 vector<**int**> Group(M);  
  
 cout << **"input Array A= { "**;  
 **for** (**int** i = 0; i <= n - 2; i++)  
 {  
 cout << sums[i] << **", "**;  
 }  
 cout << sums[n - 1] << **" }"** << **'\n'** << endl;  
  
 Group = MaxMin1(sums, n, M);  
 cout << **"Array B = { "**;  
 **for** (**int** m =  
 0; m <= M - 2; m++)  
 {  
 cout << Group[m] << **", "**;  
 }  
 cout << Group[M - 1] << **" }"** << endl;  
  
  
 **return** 0;  
}

input Array A= { 3, 9, 7, 8, 2, 6, 5, 10, 1, 7, 6, 4 }

Array B = { 3, 4, 5 }

Process finished with exit code 0