#define ll long long int

const int m = 1e9 + 7;

class Solution {

public:

int sumSubarrayMins(vector<int>& a){

stack<int> stl, str;

int i, n = a.size();

stl.push(0);

str.push(0);

ll nle[n], nre[n]; // stores index of nearest smallest to left/right

for(i=0;i<n;i++){

nle[i] = -1;

nre[i] = -1;

}

for(i=1;i<n;i++){

while(!str.empty() && a[str.top()]>=a[i]){

nre[str.top()] = i;

str.pop();

}

str.push(i);

while(!stl.empty() && a[stl.top()]>=a[i])

stl.pop();

if(stl.size() && a[stl.top()]<=a[i])

nle[i] = stl.top();

stl.push(i);

}

int ans = 0;

for(i=0;i<n;i++){

ll t; // stores no of subarrays where a[i] is minimum

if(nle[i]==-1){

if(nre[i]==-1)

t = (((n-i)\*(i+1))%m\*a[i])%m;

else

t = ((nre[i]-i)\*((i+1)\*a[i])%m)%m;

}

else{

if(nre[i]==-1)

t = ((n-i)\*((i-nle[i])\*a[i])%m)%m;

else

t = ((nre[i] - i)\*((i - nle[i])\*a[i])%m)%m;

}

ans = (ans + t)%m;

}

return ans;

}

};