题意：给定一个递增的数列，求出从这个数列中取出3个数，最大值与最小值的差值小于等于k的有多少种。

思路：有点递推的感觉，每加进来一个数，只考虑这个数的贡献，例如，加进来第5个数的时候，假设前面四个数都符合要求，那么5的贡献就是从前面4个数当中取出两个来的总数，所以就是C 4 2, 这是个组合数。所以考虑每一个数，把每个数的组合数加起来就是了

因为他给你的数据就是有序的，所以可以用二分

二分：

#include<cstdio>

#include<iostream>

#include<bits/stdc++.h>

using namespace std;

typedef long long ll;

const int INF=1000000007;

ll a[100010];

int main()

{

//freopen("input.txt","r",stdin);

int n;

ll d;

scanf("%d %I64d",&n,&d);

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

scanf("%I64d",&a[i]);

ll ans(0);

for(int i=0;i<=n-3;i++)

{

int p=upper\_bound(a,a+n,a[i]+d)-a-1;

ans+=(ll)(p-i)\*(p-i-1)/2LL;

}

printf("%I64d\n",ans);

return 0;

}

单调栈：

#include<cstdio>

#include<iostream>

#include<bits/stdc++.h>

using namespace std;

typedef long long ll;

const int INF=1000000007;

ll q[100010];

ll a[100010];

int main()

{

//freopen("input.txt","r",stdin);

int n;

ll d;

scanf("%d %I64d",&n,&d);

int head(1),tail(0);

ll ans(0);

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

{

scanf("%I64d",&a[i]);

q[++tail]=a[i];

while(head<=tail && q[head]<a[i]-d)

head++;

ll len=tail-head;

ans+=len\*(len-1)/2;

}

printf("%I64d\n",ans);

return 0;

}