**Bài báo cáo thực hành nhóm 3**

**Tóm tắt bài toán tảo biển:**

sau khi chúng ta xem xét lại yêu cầu của bài toán và vẽ ra một sơ đồ theo dõi kết quả , thì chúng ta có thể thấy được rằng bài toán này là một dạng mở rộng của tìm chuỗi fibonacy kết quả của nó chỉ đơn giản là thay đổi 1 vài chi tiết nhỏ .

**thuật toán (python)**

n,k = map(int,input().split())

def fibo(n):

f0 = 0

f1 = 1

fn = 1

if (n == 1):

return n

else:

for i in range(2, n):

f0 = f1

f1 = fn

fn = f0 + f1

return fn

print((n \* fibo(2\*k + 1)) % (10\*\*9 + 7))

Nhập n , k

(n \* fibonacci(2\*k + 1)) % (10\*\*9 + 7)

False

N<2

Kết qua

1

X0 =1

X1=1

Xn=1

i<=2 ,i<k

xn=x0+x1

x0=x1

x1=xn

i++

**Tóm tắt bài toán bot :**

Sau khi tóm tắt lại đề chúng ta có thể rút ra được nhận xét về bài toán , đề cho chúng t 1 dãy n số nguyên âm và dương , yêu cầu chúng ta phải tìm chuổi con có tổng lớn nhất (vị trí bắt đầu , kết thúc và sum của chuỗi )

Lưu ý : mỗi khi chúng ta xử dụng 1 biến sum để check tổng thì phải lưu ý xóa biến này mỗi khi sum trở về 0 ;

Ví dụ : 1,-1,-10,11,-11,5 …. Nếu chúng ta không đặt điều kiện thì kết quả in ra sẽ là 11,-11,5 vì cơ bản biến sum của 11+(-11)+5 == 5 (sai)

x = int(input())

count = [1]\*x

res = [0]\*x

arr = list(map(int, input().split()))

ans = arr[0]

res[0] = arr[0]

for i in range(1, x):

res[i] = arr[i]

count[i] = 1

if res[i-1] >= 0:

res[i] += res[i-1]

count[i] += count[i-1]

ans = max(ans, res[i])

a = 0

b = 0

for i in range(0, x):

if res[i] == ans:

a = i

b = count[i]

break

print(a+2-b, a+1, ans)

Kq

//a+2-b,a\_1,ans

A=i

B=count[i]

Break

i--

Res[i-1]==ans

A=0

B=0

I<x

Ans=max(ans,res[i]

i--

Res[i]+=res[i-1]

Count[i]+=count[i-1]

Res[i-1]>=0

Res[i]=arr[i]

Count[i]=1

I<x

Made an arr of Xth

count =[1]\*x

result=[0}\*x

ans =arr[0]

res[0]=arr[0]

Nhập x