BÀI TẬP 2: TỐI ƯU HÓA CHƯƠNG TRÌNH

Cho dãy A gồm N số nguyên dương (1≤ N ≤1000), A[1..N] và số nguyên dương X≤1000.

Yêu cầu: Hãy đề nghị giải thuật tối ưu nhất có thể về thời gian hoặc không gian để tìm dãy con gồm nhiều phần tử nhất của dãy A ở trên, sao cho tổng các phần tử của dãy con này chia hết cho X và viết hàm cài đặt.

Ví dụ :

- Input: N = 10 ; X = 5

- Mảng A1 , A2 ,..., AN = {1,6,11,5,10,15,20,2,4,9}

- Output: Độ dài dãy con tìm được = 8

- Tổng các phần tử của dãy con : 80

Danh sách các phần tử :

A[10] =9; A[9]=4; A[7]=20; A[6]=15;

A[5]=10; A[4]=5; A[3]=11; A[2]=6

Ý tưởng: Sử dụng chiến lược quy hoạch động, tận dụng bộ nhớ để lưu lại kết quả các bài toán con, tránh việc tính lại các bài toán con nhiều lần để tối ưu về thời gian.

Cách làm:

Khai báo mảng F[1001][1000] để lưu kết quả tính toán theo công thức quy hoạch động, mảng a[1001] để lưu giá trị các số vừa nhập.

Khi nhập các giá trị a[i], rút gọn giá trị xuống thành a[i] % X, vì thế mảng a[i] chỉ gồm các phần tử có giá trị trong đoạn [0; X - 1] để tối ưu bộ nhớ mà không ảnh hưởng kết quả.

Gọi F[i][j] là độ dài của dãy con dài nhất xét đến vị trí thứ i (1 <= i <= N), sao cho tổng các phần tử của nó chia lấy dư với X bằng j (0 <= j <= X - 1).

Với mỗi i, có 2 khả năng:

- Nếu không chọn a[i], F[i][j] = F[i-1][j]

- Nếu chọn a[i], F[i][j] = F[i - 1][t] + 1 với t = (j - a[i] + X) % X

Do đó ta có công thức quy hoạch động:

F[i][j] = max(F[i-1][j], F[i - 1][ (j - a[i] + X) % X] + 1)