**CÁC BÀI TRONG ĐỀ KIỂM TRA CUỐI HỌC KỲ - MÔN THUẬT TOÁN**

**BÀI 1. DÃY CON KHÔNG LIÊN TIẾP BẬC K**

Cho dãy số A[] có N phần tử là các số nguyên dương.

Với mỗi số nguyên K, một **dãy con không liên tiếp bậc K** được định nghĩa là dãy con của dãy A[] trong đó không có K số liên tiếp nào trong dãy A[] được chọn (tức là chỉ được phép chọn tối đa K-1 số liên tiếp).

Hãy tính tổng lớn nhất có thể của dãy con không liên tiếp bậc K.

**Input**

* Dòng đầu tiên ghi hai số N và K (1 ≤ N ≤ 10000; 1 ≤ K ≤ 1000)
* Dòng tiếp theo ghi N số của dãy A[], các giá trị đều nguyên dương và không quá 1000

**Output**

Ghi ra kết quả tính được trên một dòng.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5 2  10 10 20 20 10 | 40 |

**BÀI 2. DI CHUYỂN K BƯỚC**

Cho đồ thị vô hướng có N đỉnh dưới dạng ma trận kề. Các đỉnh được đánh số từ 1 đến N. Với mỗi cặp đỉnh S và T trong đồ thị, người ta muốn đếm số cách đi từ S đến T sao cho số cạnh đi qua đúng bằng K. Giả sử các đỉnh và các cạnh có thể được lặp lại nhiều lần trong đường đi.

**Input**

Dòng đầu tiên ghi 4 số N, S, T, K (1 ≤ N ≤ 50; 1 ≤ S, T ≤ N)

N dòng tiếp theo ghi ma trận kề của đồ thị.

**Output**

In ra số tuyến đường thỏa mãn chia dư cho 109 + 7.

**Subtask**

* Subtask 1 (50%): 1 ≤ K ≤ 105
* Subtask 2 (50%): 1 ≤ K ≤ 1018

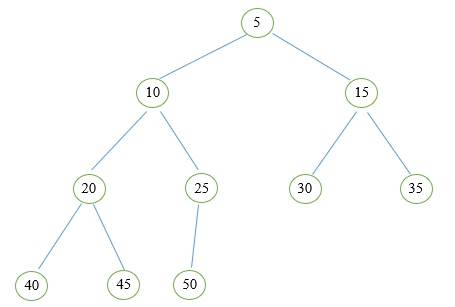
**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 1 1 4  0 1 0  1 0 1  0 1 0 | 2 |

**BÀI 3. DSA\_P212. CÂY NHỊ PHÂN GẦN ĐẦY - 2**

Cho mảng A[] có N phần tử. Lấy phần tử A[0] làm gốc của cây, xây dựng cây nhị phân gần hoàn chỉnh với các phần tử còn lại A[1], A[2], …, A[N-1]. Nhập lần lượt từng tầng của cây, mỗi tầng gồm có 1, 2, 4, 8, … phần tử.

Ví dụ với mảng A[] = {5, 10, 15, 20, 25, 30, 35, 40, 45, 50}, ta xây dựng được cây nhị phân như sau:



Hãy in ra kết quả phép duyệt **postorder** của cây nhị phân gần hoàn chỉnh này.

**Input:**

Dòng đầu tiên là số nguyên bộ test T (T ≤ 100).

Mỗi test bắt đầu số nguyên N (1 ≤ N ≤ 10 000).

Dòng tiếp theo gồm N số nguyên A[i] (0 ≤ A[i] ≤ 10 000).

**Output:**

Với mỗi test, in ra kết quả của phép duyệt **postorder**trên một dòng.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  10  5 10 15 20 25 30 35 40 45 50  10  1 2 3 4 5 6 7 8 9 10 | 40 45 20 50 25 10 30 35 15 5  8 9 4 10 5 2 6 7 3 1 |

**BÀI 4. CHÊNH LỆCH LỚN NHẤT**

Cho một xâu S gồm N kí tự in thường (từ ‘a’ đến ‘z’). Bạn muốn chọn ra một xâu con liên tiếp (khác rỗng) từ S, sao cho chênh lệch giữa số lần xuất hiện của kí tự xuất hiện nhiều nhất và kí tự xuất hiện ít nhất là cực đại. Nói cách khác, nếu gọi A là số lần xuất hiện của kí tự xuất hiện nhiều nhất, và B là số lần xuất hiện của kí tự xuất hiện ít nhất, bạn mong muốn A – B đạt giá trị cực đại (giá trị này có thể bằng 0, nếu xâu bạn chọn chỉ gồm một loại kí tự).

**Input**

Dòng 1 ghi số N là số ký tự trong xâu (1 < N ≤ 106)

Dòng 2 ghi N ký tự của xâu, các ký tự đều là chữ cái viết thường.

**Output**

Ghi ra giá trị chênh lệch cực đại tìm được.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10  aabbaaabab | 3 |

*Giải thích: xâu con aaaba có 4 kí tự a và 1 kí tự b, có chênh lệch là 3.*

**BÀI 5. DI CHUYỂN ROBOT – 2**

Trên sân thi đấu được có giới hạn 105 x 105 , các chú robot không còn sử dụng công nghệ dò đường line nữa mà thực hiện di chuyển bằng các bước nhảy trên N vị trí đặc biệt và có hỗ trợ trạm sạc nhiên liệu. Robot chỉ được nhảy từ trạm nhiên liệu (x1, y1) tới (x2, y2) nếu:

* y2 = y1 và x2 > x1 hoặc,
* x2 = x1 và y2 > y1

Ban đầu robot xuất phát tại địa điểm 1, được nạp năng lượng tại địa điểm này, sau đó bắt đầu thực hiện các bước nhảy. Mỗi bước nhảy sẽ tốn mất K đơn vị nhiên liệu. Robot sẽ không thực hiện được bước nhảy nếu như không đủ nhiên liệu.

Nhiệm vụ của bạn là hãy lên một hành trình tối ưu sao cho sau khi nhảy tới địa điểm thứ N, robot vẫn còn nhiều nhiên liệu nhất có thể.

Input luôn được đảm bảo để có đáp số.

**Input:**

Dòng đầu tiên chứa hai số nguyên N và K (1 ≤ K ≤ 1000).

N dòng tiếp theo, dòng thứ i chứa 3 số nguyên X, Y, T (0 ≤ T ≤ 1000), biểu diễn trạm sạc nhiên liệu thứ i có tọa độ (X,Y) và có chứa T đơn vị nhiên liệu. Sẽ không có hai trạm nào bị trùng tọa độ.

**Output:**

In ra một số nguyên là lượng nhiên liệu lớn nhất mà robot có được sau khi tới vị trí N.

**Giới hạn:**

Subtask 1 (40%): 2 ≤ N ≤ 2000, 0 ≤ X, Y ≤ 2000.

Subtask 2 (60%): 2 ≤ N ≤ 300 000, 0 ≤ X, Y ≤ 100 000.

**Example:**

|  |  |
| --- | --- |
| Input | Output |
| 6 5  1 1 5  2 1 5  1 2 4  2 3 5  3 2 30  3 3 5 | 5 |
| 8 10  1 1 15  2 2 30  1 2 8  2 1 7  3 2 8  2 3 7  4 2 100  3 3 15 | 36 |

Giải thích test 1: Hành trình (1, 1) à (2, 1) à (2, 3) à (3, 3)

**BÀI 6. CHỌN TẬP HỢP TRÊN CÂY**

Cho một cây gồm N nút, nút thứ i được gán nhãn trọng số bằng C[i]. Hãy chọn ra một tập hợp các nút trên cây, sao cho không có hai nút kề nhau được chọn (2 nút có cạnh nối trực tiếp) và tổng trọng số của tập hợp thu được là lớn nhất.

**Input:**

Dòng đầu tiên là số nguyên dương N (N ≤ 100 000).

Dòng tiếp theo gồm N số nguyên mô tả trọng số C[i] (-1000 ≤ C[i] ≤ 1000)

N-1 dòng tiếp, mỗi dòng gồm 2 số nguyên u, v mô tả cạnh nối uàv.

**Output:**

In ra tổng trọng số của tập hợp tìm được.

|  |  |
| --- | --- |
|  |  |

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  1 -2 2 3 -1  1 2  2 3  2 4  1 5 | 6 |
| 5  1 -2 2 3 8  1 2  2 3  2 4  1 5 | 13 |

Giải thích test 1: Chọn {3, 4, 1}

Giải thích test 2: Chọn {3, 4, 5}