**15. 3Sum [Array]**

***Câu hỏi:***

Given an integer array nums, return all the triplets [nums[i], nums[j], nums[k]] such that i != j, i != k, and j != k, and nums[i] + nums[j] + nums[k] == 0.

Notice that the solution set must not contain duplicate triplets.

***Phân tích:***

**Input:** 1 mảng số nguyên a

**Process:**

* Bộ 3 số cộng lại = 0 và các bộ 3 không trùng nhau

**Output:**

* Mảng 2 chiều

***Solution***

* **Sort**
* Vòng lặp for  i : 0 -> n-2 ( từ đầu-trước k)
* Vòng lặp while để tìm  j= i+ (kế i);  k = n-1 (cuối cùng)
* i cố định để tìm j, k đến khi vòng lặp while kết thúc thì mới tăng i rồi tìm tiếp như ban đầu
* i > 0 && a[i] === a[i-1] -> continue: bỏ qua câu lệnh và chạy vòng lặp mới khi trùng lặp
* J < k: tính sum
* Sum === 0: -> (a[i], a[j], a[k])
* j < k && a[j] === a[j+1] -> j++ : khi bị trùng thì chạy tiếp bỏ qua phần tử trùng
* j < k && a[k] === a[k-1] -> k–
* -> j++, k–
* Sum < 0 : j++
* Sum > 0: k–