Drzewa i słowniki e02

Zadanie 1.
Dana jest tablica z haszowaniem T[N], w której elementy były wstawiane za pomocą funkcji haszującej hash(x) z użyciem liniowego rozwiązywania kolizji. W tej tablicy -1 oznacza wolne miejsce. Proszę podać jak najszybszy algorytm sprawdzający, czy dana tablica jest poprawna - czyli, czy każdy element może być odnaleziony na podstawie swojego hasha.

Zadanie 2.
Dany jest zbiór n punktów na płaszczyźnie dwuwymiarowej, w postaci tablicy par liczb. Podaj jak najszybszy algorytm zwracający rozmiar największego podzbioru tych punktów, takiego, że wszystkie są współliniowe.

Zadanie 3.
Dana jest struktura:
class Node:
    nodes # lista referencji na sąsiadów
reprezentująca węzeł grafu. Podaj algorytm, który mając listę referencji na obiekty klasy Node, stworzy kopię grafu reprezentowanego przez listę, tak, aby kopia miała dokładnie taką samą strukturę. Referencja z punktu widzenia komputera jest adresem pamięci, a więc zwykłą liczbą, co oznacza, że może być haszowana jak zwykły int.

Zadanie 4.
Dana jest T[N] zawierająca liczby całkowite, również ujemne. Podaj algorytm, który sprawdzi, czy jest w tej tablicy spójna podtablica, której elementy sumują się do zadanej liczby S.

Zadanie 5.
Dane jest słowo S, oraz lista słów W, w której wszystkie słowa mają tę samą długość i mogą się powtarzać. Podaj algorytm, który znajdzie wszystkie takie indeksy i, że na pozycji S[i] zaczyna się substring, będący konkatenacją wszystkich słów W, w dowolnej kolejności i z taką samą liczbą wystąpień każdego słowa z jaką występuje ono w W.
Przykład:
S = ‘barfoothefoobarman’
W = [‘foo’, ‘bar’]
Output = [0,9]

Zadanie 6.
Dane są dwie tablice z haszowaniem reprezentujące zbiory mnogościowe, co do których podejrzewamy, że zawierają dużą liczbę kolizji. Podaj algorytm, który znajdzie część wspólną tych zbiorów, w czasie pesymistycznym O(nlog(n)) i czasie oczekiwanym O(n).
W związku z powyższym algorytm ma działać w oczekiwanym czasie O(n), ale w przypadku dużej liczby kolizji ma działać w O(nlog(n)), a nie w O(n^2), co oznacza że nie

akceptujemy rozwiązania w którym dla każdego elementu jednej tablicy szukamy, czy znajduje się on w drugiej. Niepoprawne będzie również przepisanie tablic do drzew, ponieważ to dałoby zawsze O(nlog(n)).

Zadanie 7.
Dana jest tablica stringów T. Podaj algorytm, który znajdzie wszystkie takie pary (i,j), że konkatenacja T[i]+T[j] jest palindromem.

Zadanie 8.
Dany jest zbiór S punktów z przestrzeni d - wymiarowej. Zaproponuj strukturę danych, która pozwala szybko odpowiadać na pytania czy jakiś d - wymiarowy punkt P, znajduje się w S.