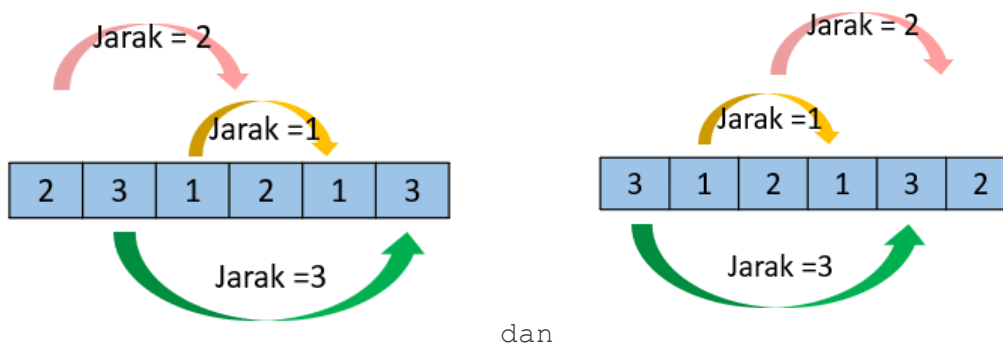


BACKTRACKING

Womi sedang mengikuti lomba matematika di sekolahnya. Womi meminta bantuan kalian sebagai anggota timnya membuat program yang dapat menghitung banyaknya kombinasi yang mungkin untuk membuat untaian yang terdiri dari $2n$ bilangan di mana setiap bilangan $[1..n]$ muncul tepat dua kali dalam untaian tersebut. Syaratnya adalah jarak dari setiap bilangan yang sama harus tepat sebesar nilai bilangan itu sendiri. Contoh: jika $n=3$, banyaknya kombinasi yang mungkin adalah 2, yaitu untaian:



Buatlah program untuk menyelesaikan persoalan ini dengan teknik *backtracking*!

Spesifikasi masukan

Masukan berupa untaian sebuah bilangan bulat n yang menyatakan bilangan terbesar dalam untaian.

Spesifikasi keluaran

Keluaran program berupa:

- m baris berisi untaian pasangan bilangan $1..n$ yang memenuhi persyaratan: jarak antar bilangan yang sama adalah sebesar bilangan itu sendiri. Setiap untaian diawali dan diakhiri dengan tanda kurung siku. Masing-masing elemen dipisahkan koma.
- bilangan bulat m yang menyatakan banyaknya untaian berbeda yang mungkin terbentuk dengan memenuhi persyaratan.

Contoh masukan

3

Contoh keluaran

[3, 1, 2, 1, 3, 2]

[2, 3, 1, 2, 1, 3]

2

Hint

Untuk menyelesaikan persoalan ini, kamu dapat memodifikasi *method* permutasi seperti yang sudah dijelaskan di kelas. Gunakan fungsi rekursif **void generateSolution(arr[], x, n)** dengan

- arr[] digunakan untuk menyimpan untaian bilangannya,
- x=[1..n] menyatakan bilangan, dan
- n adalah nilai bilangan terbesar dalam untaian.

Backtracking dilakukan untuk kembali ke keadaan sebelum suatu posisi diisi dengan nilai x.