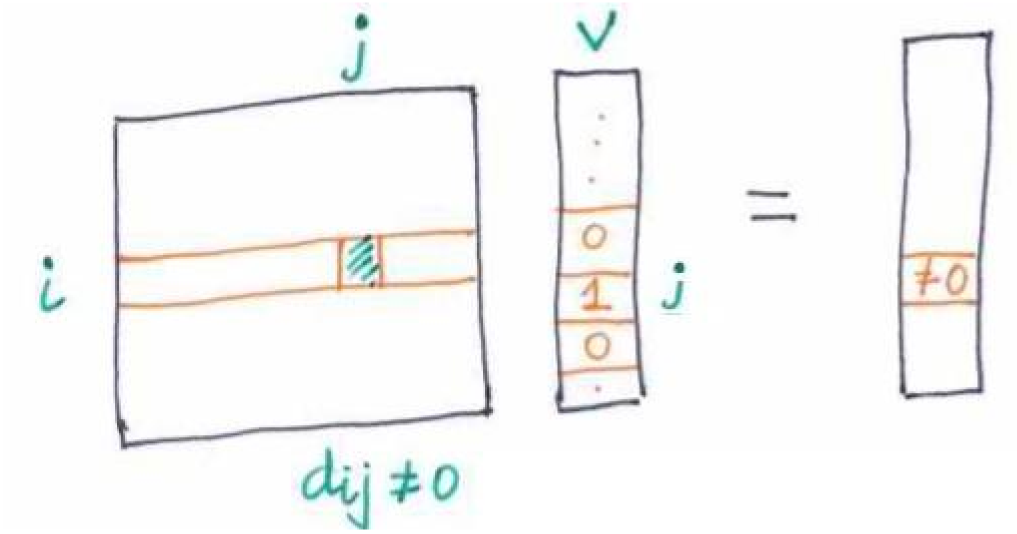
Curs 6 Slide 26.
Observatie 1: Daca AxB=C, atunci algoritmul returnează mereu “DA”
A(Br)=(AB)r=Cr - deci algoritmul va returna “DA”

Afirmatie 2: **Daca AB!=C, atunci Prob[ABr!=Cr]≥1/2**
fie D=AB-C. Ipoteza de lucru spune ca AB!=C, deci D!=$O_{n,n}$. Deci cu siguranta exista vectori *r* astfel incat Dr!=O. Scopul este sa aratam ca exista o multitudine de astfel de valori pentru *r*. Mai exact, vom arata ca Prob[Dr!=O]≥1/2

Vom arata ca pentru fiecare r, cu proprietatea ca Dr=O, exista un r’ “croit” pentru r astfel incat Dr’!=O.
Daca D!=O, exista i,j astfel incat $d_{i,j}$!=0. Alegem v - un vector de lungime *n* cu toate elementele 0, mai putin elementul de pe pozitia j.

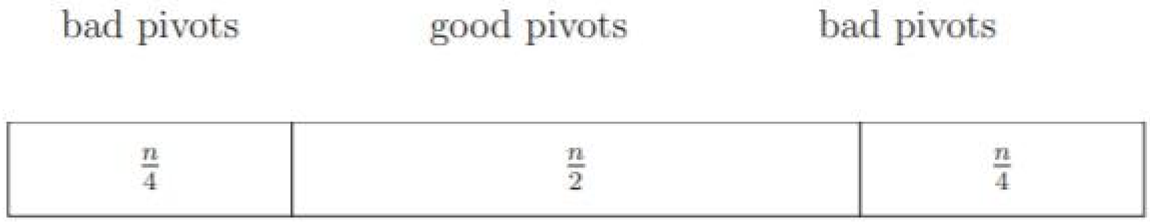


Astfel putem afirma cu siguranta ca Dv!=O.
Fie r un vector astfel incat Dr=O
Obtinem r’=r+v; r’ va fi identic cu r pe toate pozitiile, mai putin pe pozitia j, unde va fi (rj+vj) mod 2 - adica valoarea complementara.

Rezulta ca Dr’!=O. Am demonstrat ca pentru orice r exista un r’ “specific” creat astfel incat Dr’!=O
Pr[Dr!=O]≥1/2 pentru un r generat aleator

Slide 30
relatia de recurenta worst case pt basic quicksort:
$$T(n) = T(0) + T(n-1) + \theta(n)$$
$$= \theta(1) + T(n-1) + \theta(n)$$
$$= \theta\left(n^2\right)$$
Slide 38: Paranoid quick sort

Ce este un pivot “bun”?
acela pentru care partitiile L si G nu depasesc (¾)*n.
Un pivor slab este acela pentru care ori L, ori G depășește ca dimensiune valoarea ¾ *n



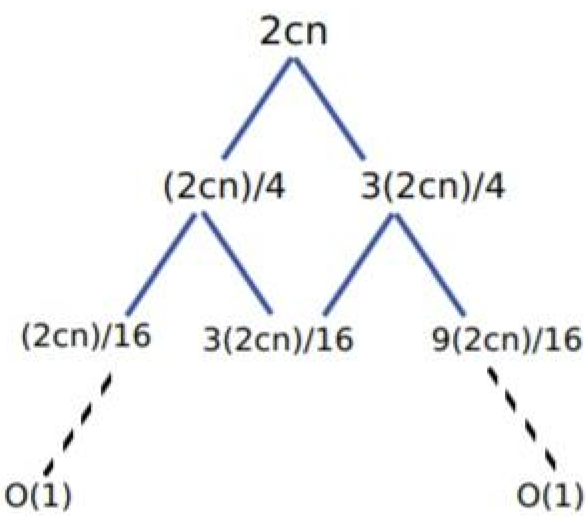
Alegand aleator un pivot, ce probabilitate este ca acel pivot ales sa fie “bun”?
½
Daca la fiecare pas al quicksort trebuie sa repet alegerea unui pivot pana cand nimeresc unul bun, in medie cate selectii trebuie facute? 2 selectii
Fie T(n) un upper bound pt numarul de pasi necesari in paranoid quicksort.
T(n) este compus din:

- numarul de pasi necesar pentru a sorta partitia L
- numarul de pasi necesar pentru a sorta partitia G
- numrul de iteratii necesar pentru alegerea pivotului si partitionarea finala dupa un pivot “bun” (nr de iteratii)*c*n

$$T(n) \leq max_{\frac{n}{4} \leq i \leq \frac{3}{4}n} \left(T(i) + T(n-i) \right) + (nt\ de\ iteratii\ pt\ alegerea\ pivotului) \cdot cn$$

$$nr\ iteratii\ = 2$$

$$T(n) \leq T\left(\frac{n}{4}\right) + T\left(\frac{3}{4}n\right) + 2cn$$



Inaltimea arborelui de derivare nu poate fi mai mult decat $\log_{4/3}(2cn)$.
$T(n) = \theta(n \log n)$