The Probabilistic Method

Week 11: Applications, Randomized Algorithms

Joshua Brody

CS49/Math59

Fall 2015

Reading Quiz

What resource is most important to optimize in streaming algorithms?

(A) runtime

(B) amount of randomness used

(C) amount of memory used

(D) tightness of approximation factor

(E) none of the above

Reading Quiz

What resource is most important to optimize in streaming algorithms?

(A) runtime

(B) amount of randomness used

(C) amount of memory used

(D) tightness of approximation factor

(E) none of the above

Error-Correcting Codes

[Shannon 48]

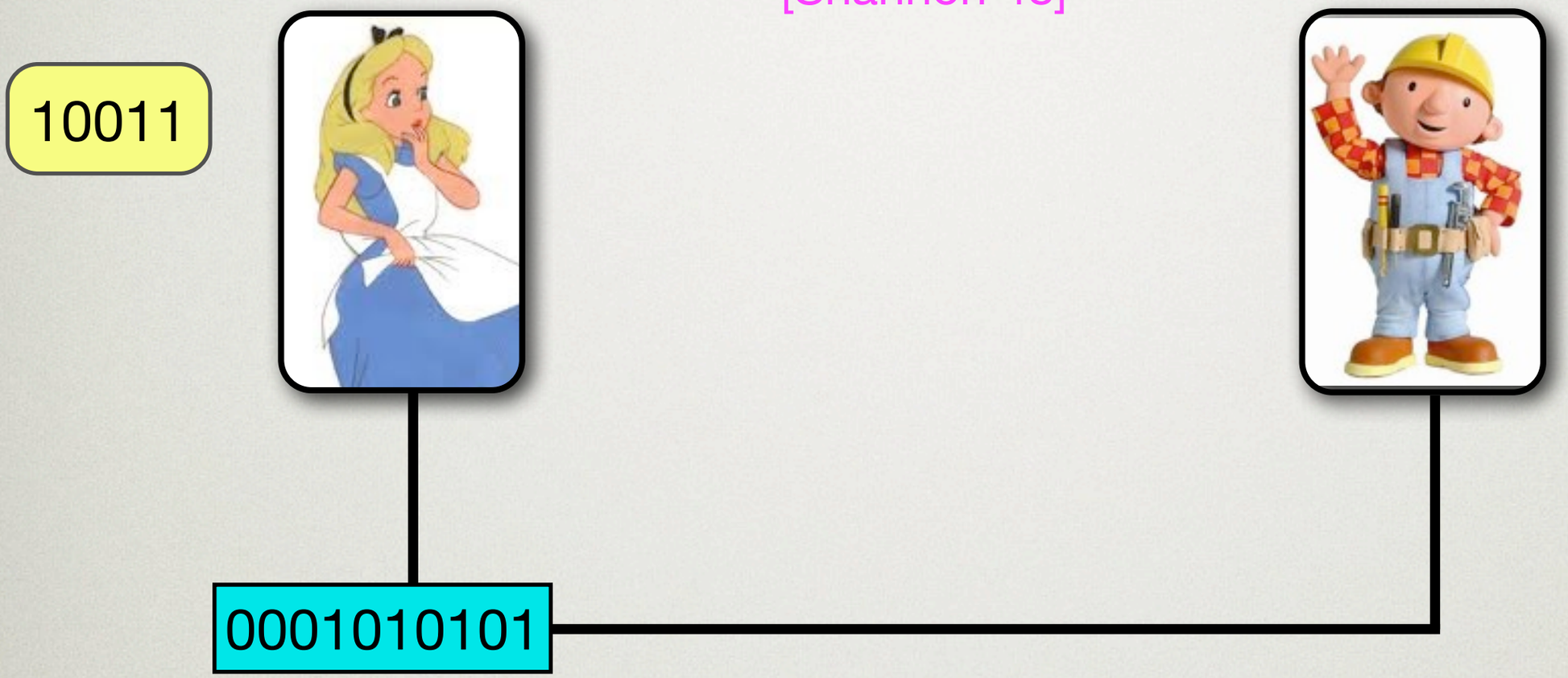
10011

ECC: set **C ⊆ {0,1}^{n}**, bijection **ENC: {0,1}^{k} → C**

- rate **R := k/n**
- distance: min **HD(c_1,c_2)**

Error-Correcting Codes

[Shannon 48]

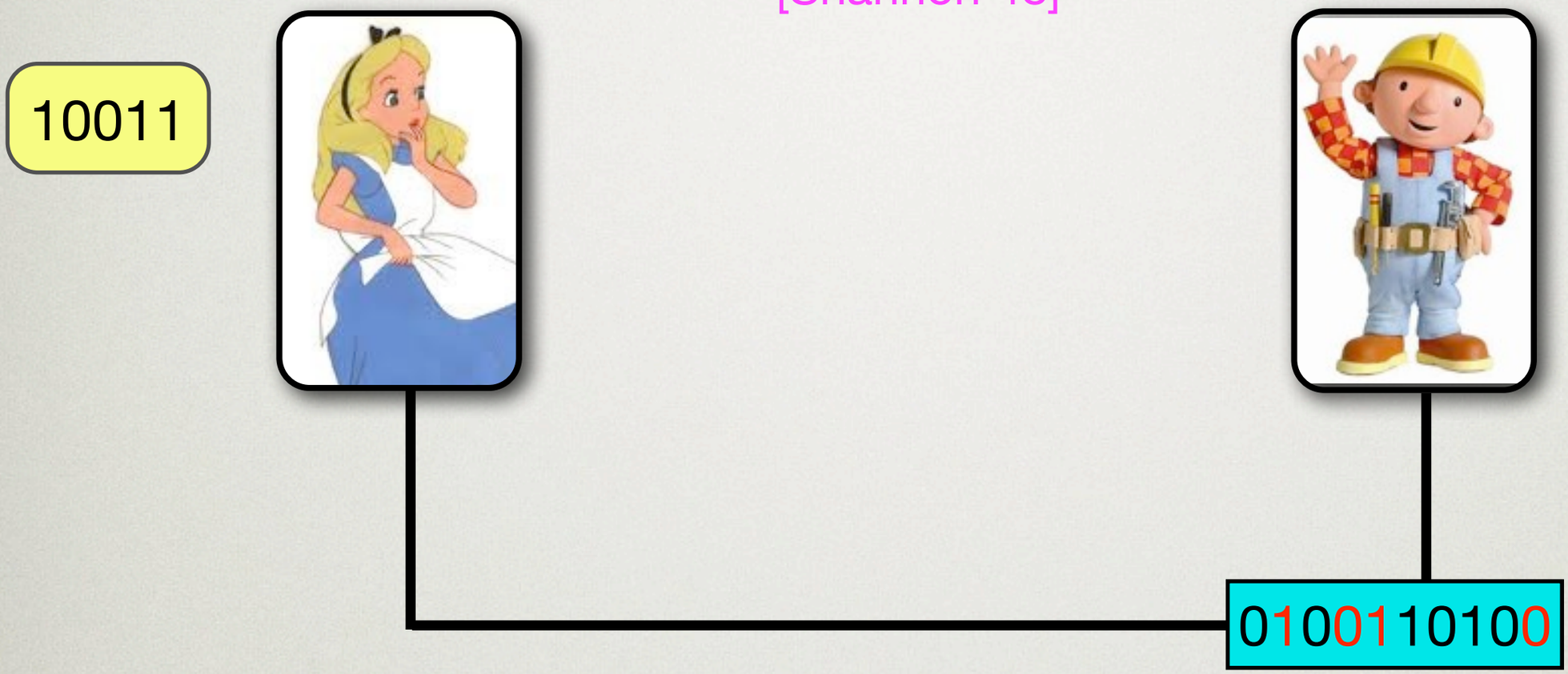


ECC: set $C \subseteq \{0,1\}^n$, bijection $ENC: \{0,1\}^k \to C$

- rate $R := k/n$
- distance: min $HD(c_1,c_2)$

Error-Correcting Codes

[Shannon 48]

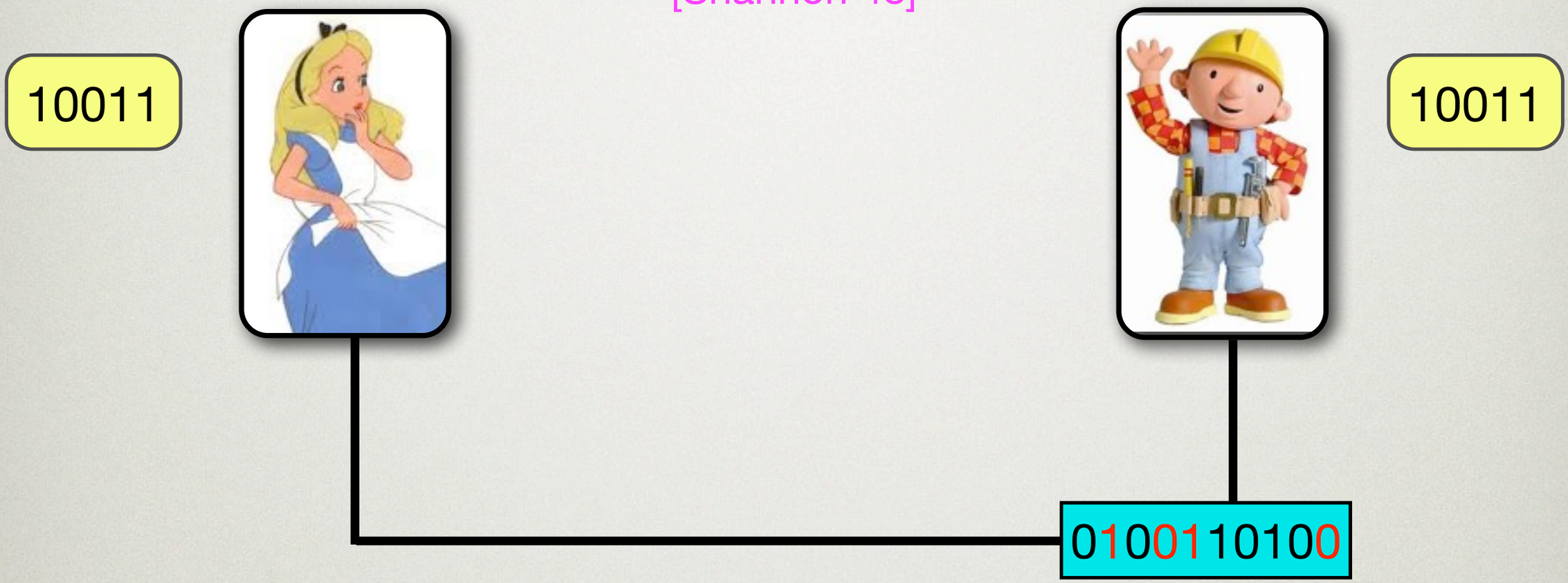


ECC: set $C \subseteq \{0,1\}^n$, bijection $ENC: \{0,1\}^k \to C$

- rate $R := k/n$
- distance: min $HD(c_1,c_2)$

Error-Correcting Codes

[Shannon 48]

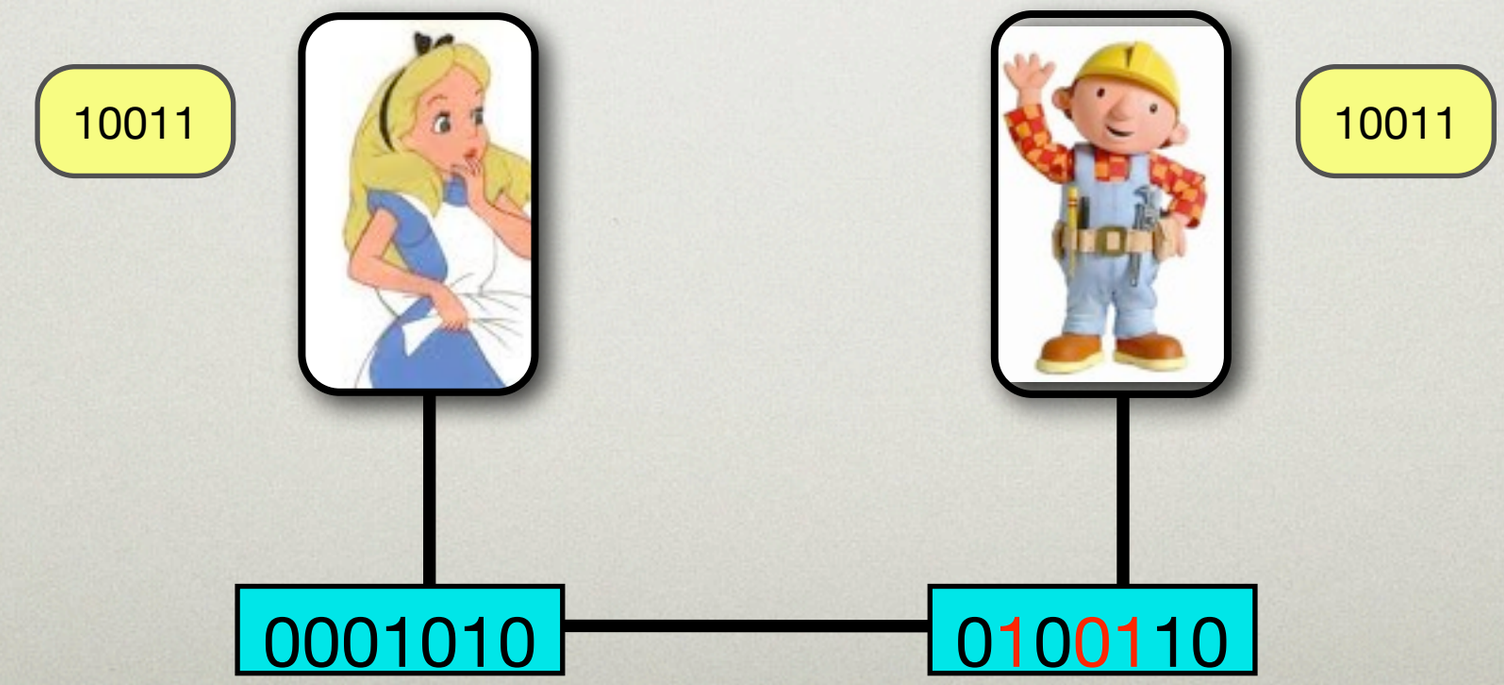


ECC: set $C \subseteq \{0,1\}^n$, bijection $ENC: \{0,1\}^k \to C$

- rate $R := k/n$
- distance: min $HD(c_1,c_2)$

Errors in ECCs

- each bit flipped w/probability **p**
- **p**-fraction of bits flipped
  - random fraction of bits flipped
  - adversarially chosen fraction of bits flipped
  - bits erased w/probability **p**
  - bits transposed w/probability **p**...



Adjacency Matrix of Magic Graphs

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}

Adjacency Matrix of Magic Graphs

columns indexed by vertices in L

rows indexed by vertices in R

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}

Adjacency Matrix of Magic Graphs

$a_{ij} = 1$ *iff* (v_j, r_i) *is edge in magic graph*

columns indexed by vertices in L

rows indexed by vertices in R

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}

Magic Graph
Matrix Multiplication

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}

$*$

x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8

$=$

y_1
y_2
y_3
y_4
y_5
y_6

Magic Graph
Matrix Multiplication

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}

$*$

x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8

$=$

y_1
y_2
y_3
y_4
y_5
y_6

$$y_i = \sum_k a_{ik} * x_k$$

Magic Graph
Matrix Multiplication

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}

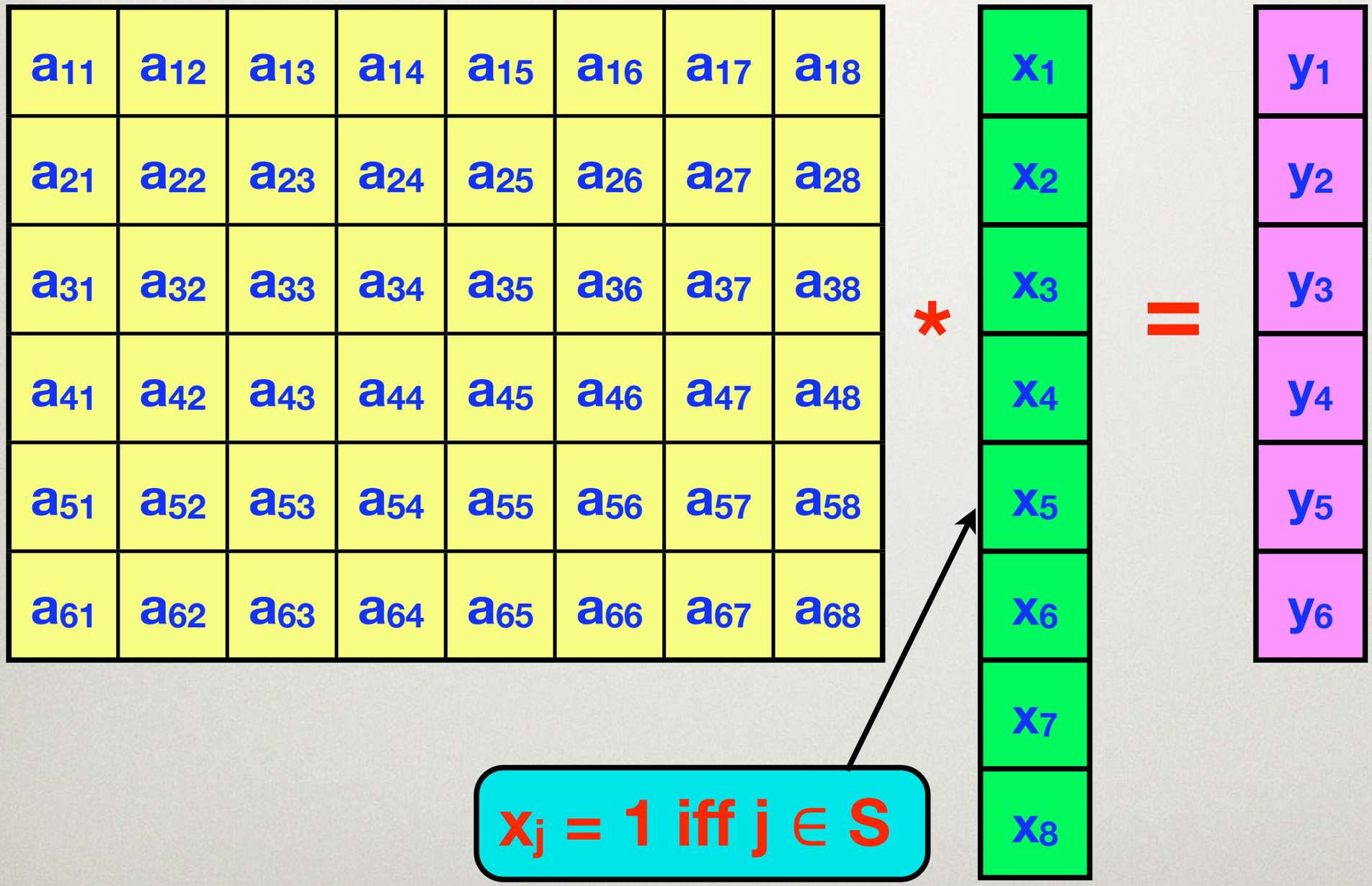
$*$

x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8

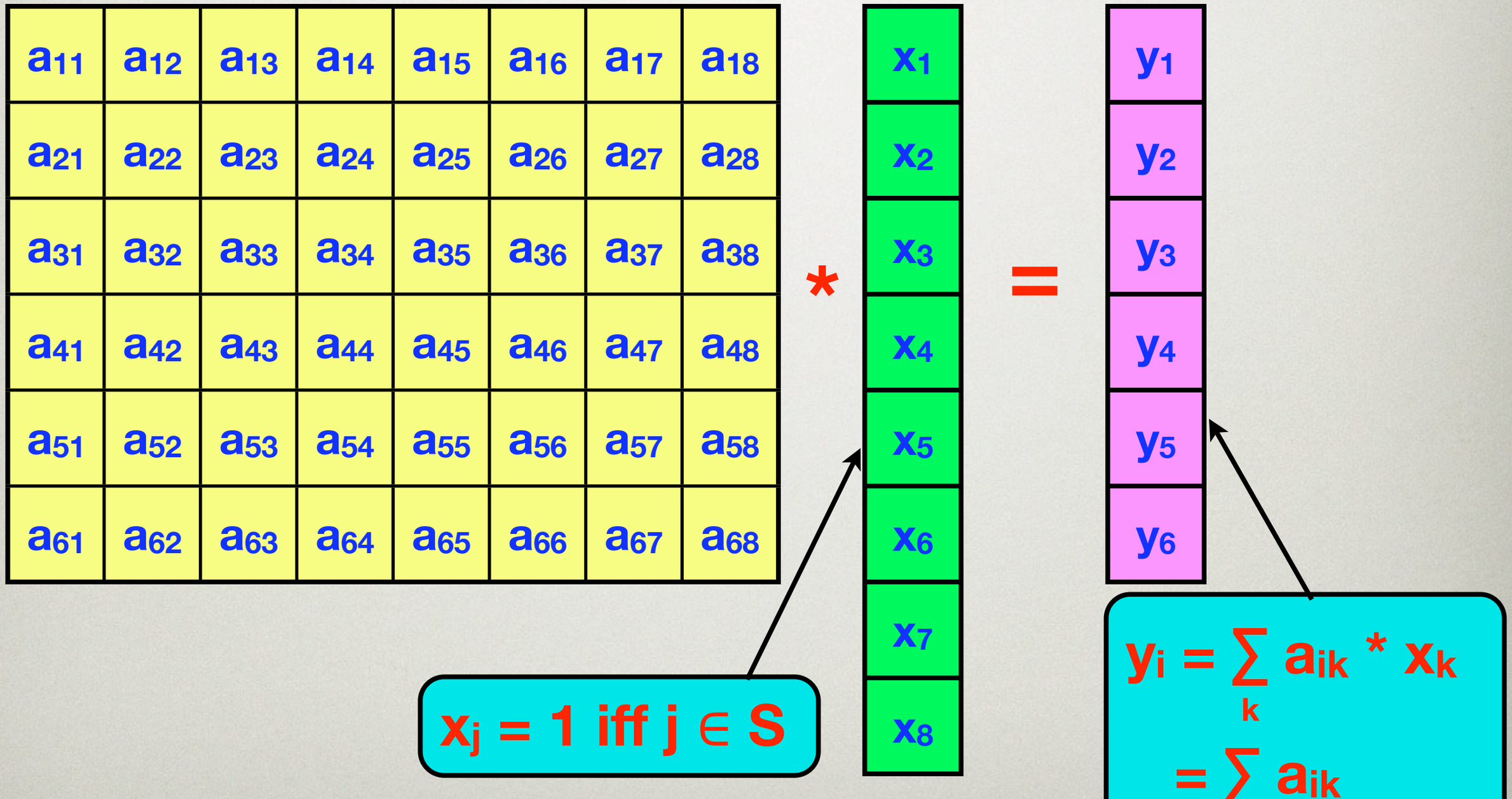
$=$

y_1
y_2
y_3
y_4
y_5
y_6

Magic Graph
Matrix Multiplication



Magic Graph
Matrix Multiplication



QUICKSORT

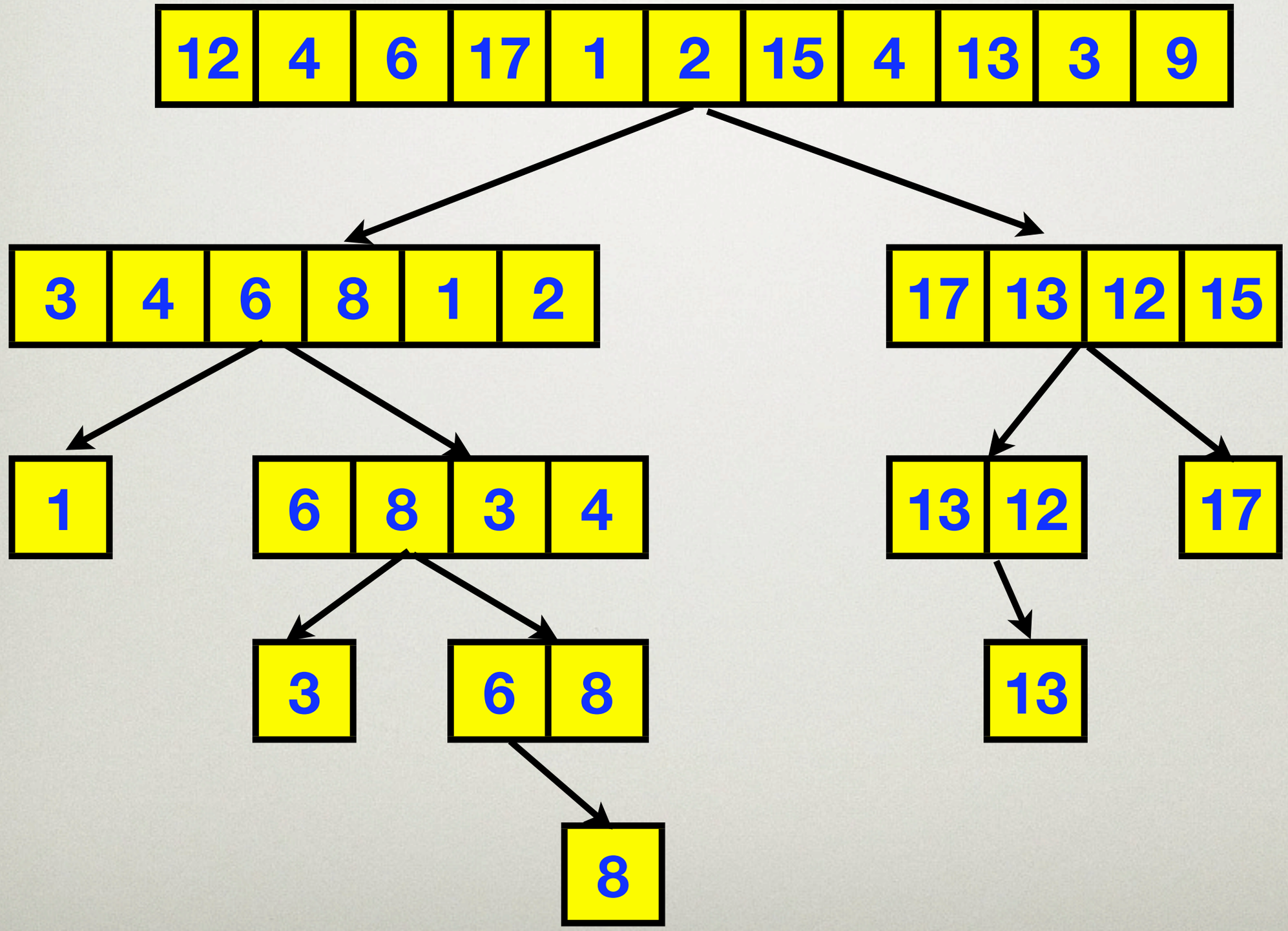
```
Algorithm quickSort(A, left, right):
  Input: array of integers A, indices left,right
  Output: nothing, but A[left...right] sorted.

  if(right > left) {
    p = partition(A, left, right)
    quickSort(A, left, p-1)
    quickSort(A, p+1, right)
  }
  Return;
}
```

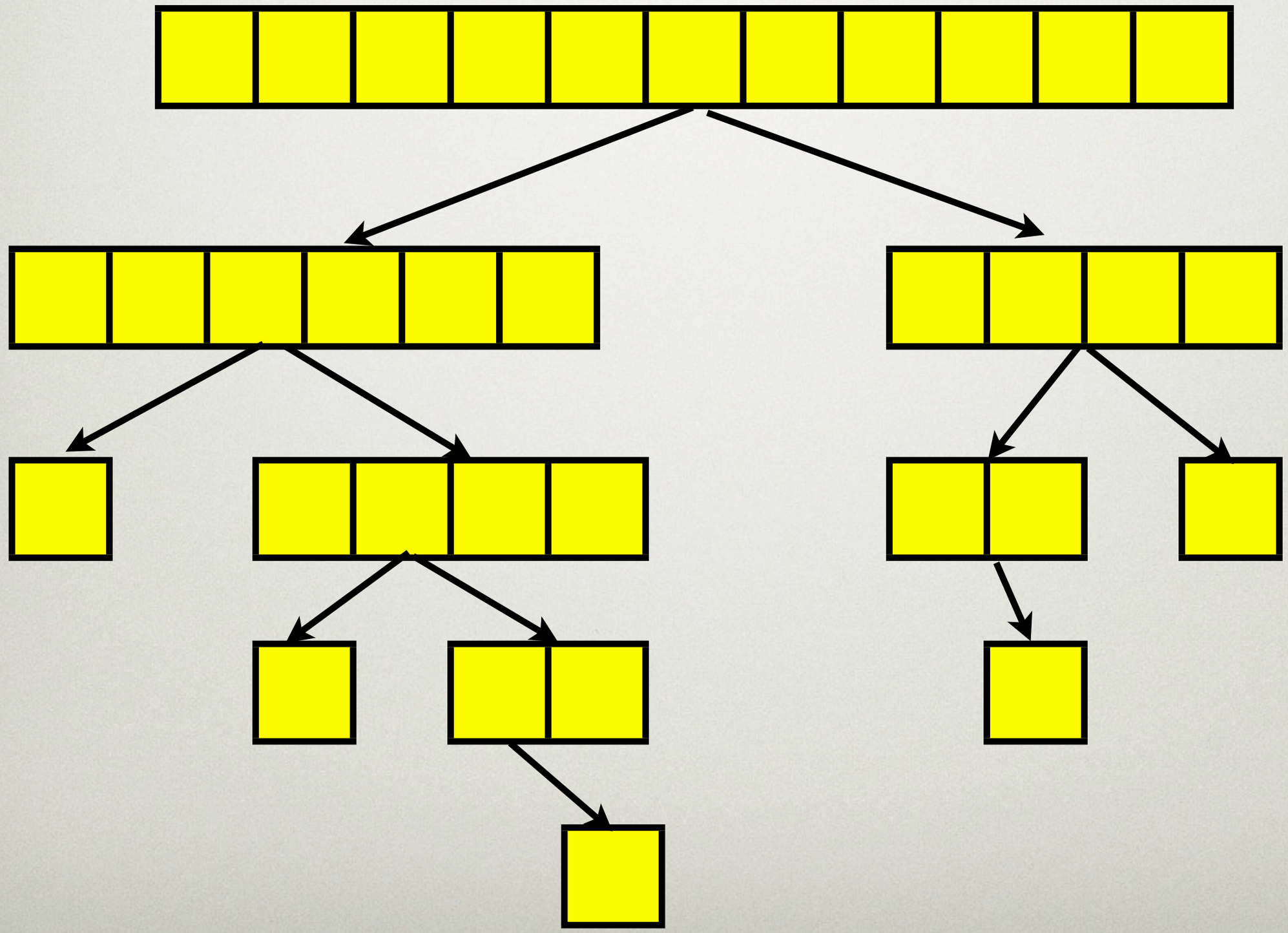
QUICKSORT

```
Algorithm partition(A, start, end):
  Input: array of integers A, range start, end
  Output: location of pivot element,
          subrange A[left...right] partitioned.
  pivot = A[end]
  left=start; right=end-1;
  while(right > left) {
    if(A[left]>=pivot) { left++; }
    if(A[right] < pivot) { right--; }
    swap(A,left,right)
  }
  swap(A,left,end)
}
```

Quicksort



Randomized Quicksort



The Probabilistic Method