Sorting

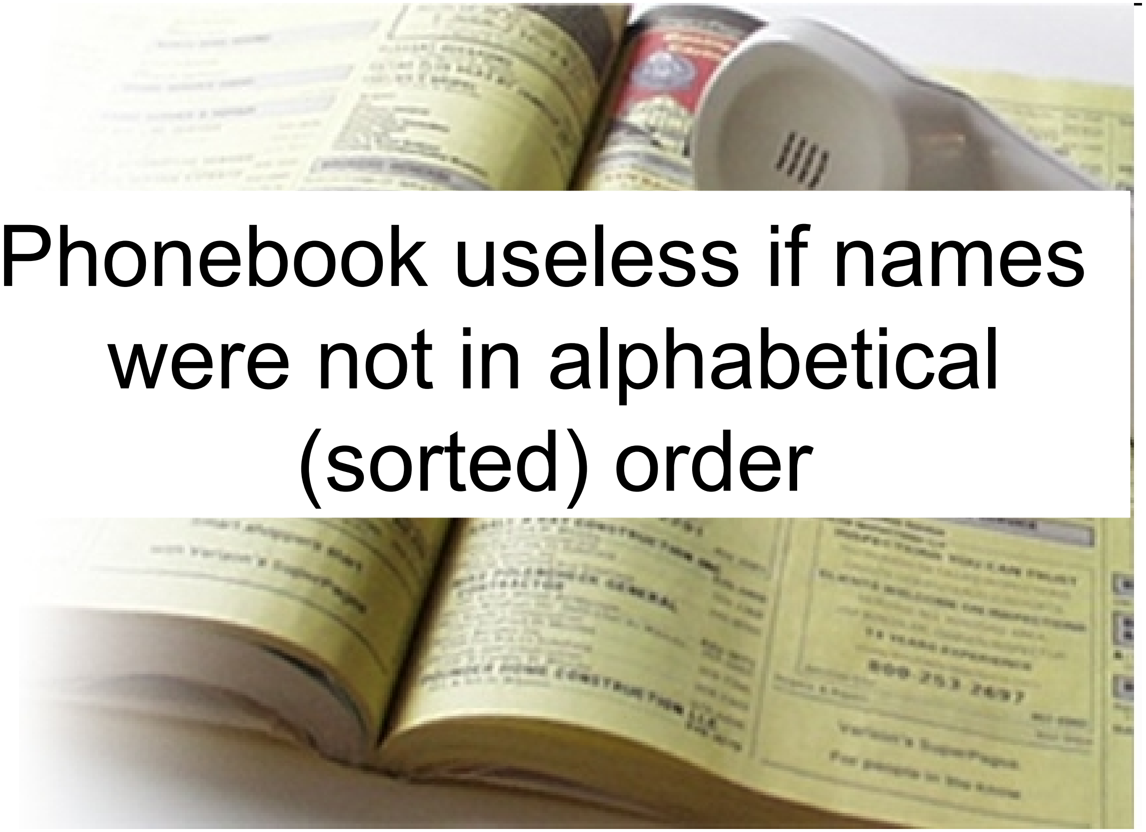
APS105: Computer Fundamentals

Jason Anderson

Dept. of Electrical and Computer Engineering
Faculty of Applied Science and Engineering
University of Toronto

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Sorting



Phonebook useless if names were not in alphabetical (sorted) order

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Why Sorting?

- Sorting used in **many** computer programs:
  - iPod sorts songs based on title or artist.
  - Facebook sorts friends in alphabetical order.
  - Facebook gives you the most-recent status.
  - Excel spreadsheet can sort column by values.
  - Online dating: sort dating matches based on how close by they live.
  - Youtube sorts based on popularity.
  - ….

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Sorting Objective

- Given: list of items (numbers, names, etc).
- Want to put the items sorted order.
    - Alpha order
    - Largest-to-smallest
    - Smallest-to-largest
    - Darkest colour to lightest colour
- Sometimes there are so many items, we need computer's help to do the sorting!

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

What is an Algorithm?

- Set of steps for completing a task.
- You have **already** been designing algorithms in this class.
- You **already** use algorithms all the time in your daily life!

Algorithm for Baking a Cake

- Cream the butter.
- Mix the dry ingredients separately.
- Combine the dry and wet ingredients.
- Mix until smooth.
- Put into baking pan.
- Bake for 30 mins at 350º F.
- Is cake done?
  - If yes, remove from oven.
  - If no, bake for another 5 minutes.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Algorithm Efficiency

- Washer, dryer take 30 mins each.
- Have one load of light clothes, one load dark clothes.

<u>Algorithm 1:</u>

2 pm : Light clothes into washer

2:30 pm: Light clothes into dryer

3 pm: Darks into washer

3:30 pm: Darks into dryer

All done at 4 pm!

<u>Algorithm 2:</u>

2 pm: Light clothes into washer

2:30 pm: Light clothes into dryer AND Darks into washer

3 pm: Darks into dryer

All done at 3:30 pm!

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Sorting Algorithms

- Sequence of steps computer takes for getting items into the right (sorted) order.
- Many different sorting algorithms invented:
  - But… they have different efficiencies!
    - Some take more “steps” to get things into the right order.
    - Some work better for different types of items.
- Want fast sorting algorithms:
  - Sort GTA phonebook with 6 million names.

Sorting Algorithms

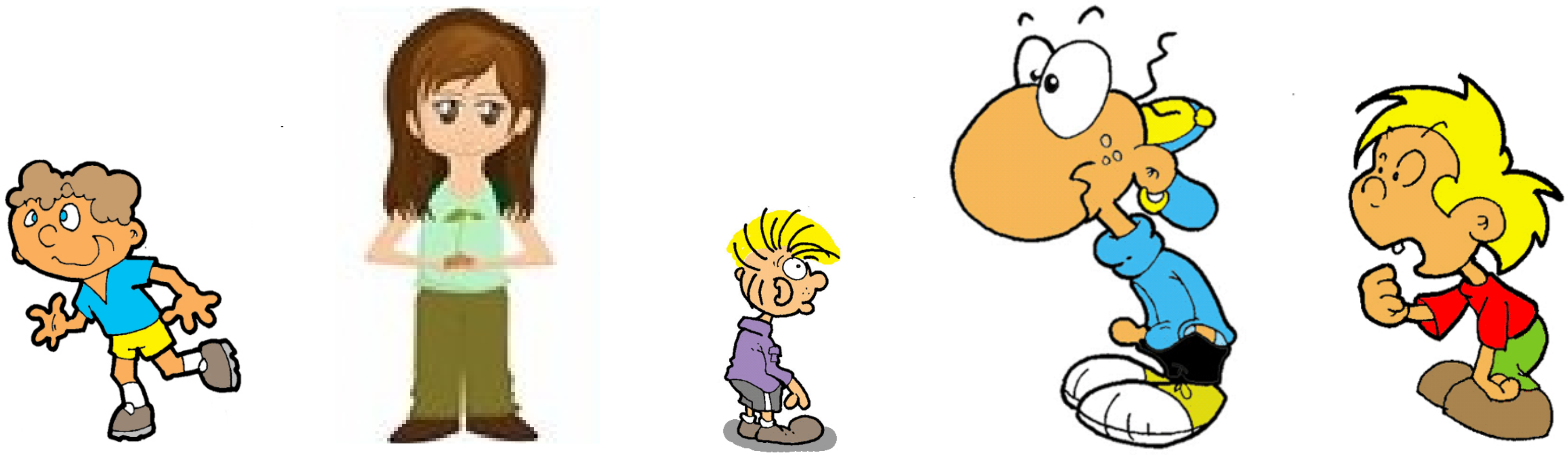
- Research on sorting algorithms:
  - Started in 1950s
  - Still active today
    (new algorithm invented in 2004)!
- We'll discuss three **classic** sorting approaches today:
  - Bubble sort
  - Insertion sort
  - Quicksort

Bubble Sort

- Walk through the list of numbers, comparing two items at a time.
  - Swap the two items if they're out of order.
- The biggest item "bubbles" up to the top.
- Walk the list of numbers several times until completely sorted!

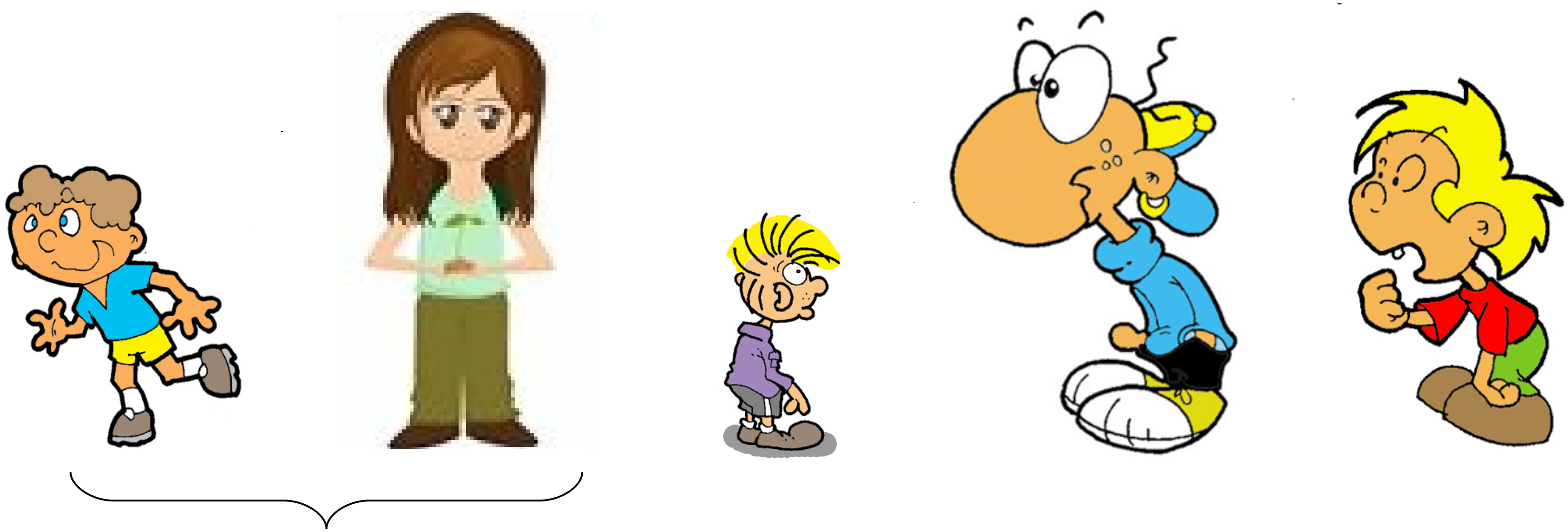
Bubble Sort

Sort Children from Shortest to Tallest



Bubble Sort

Sort Children from Shortest to Tallest

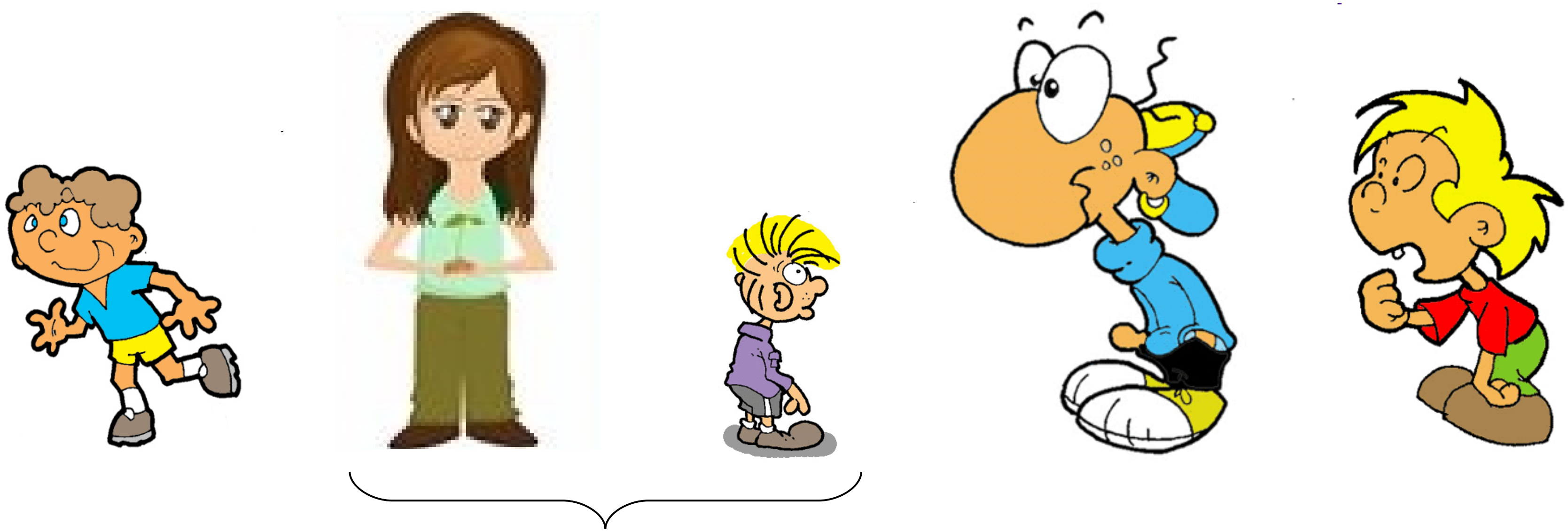


Look at first two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

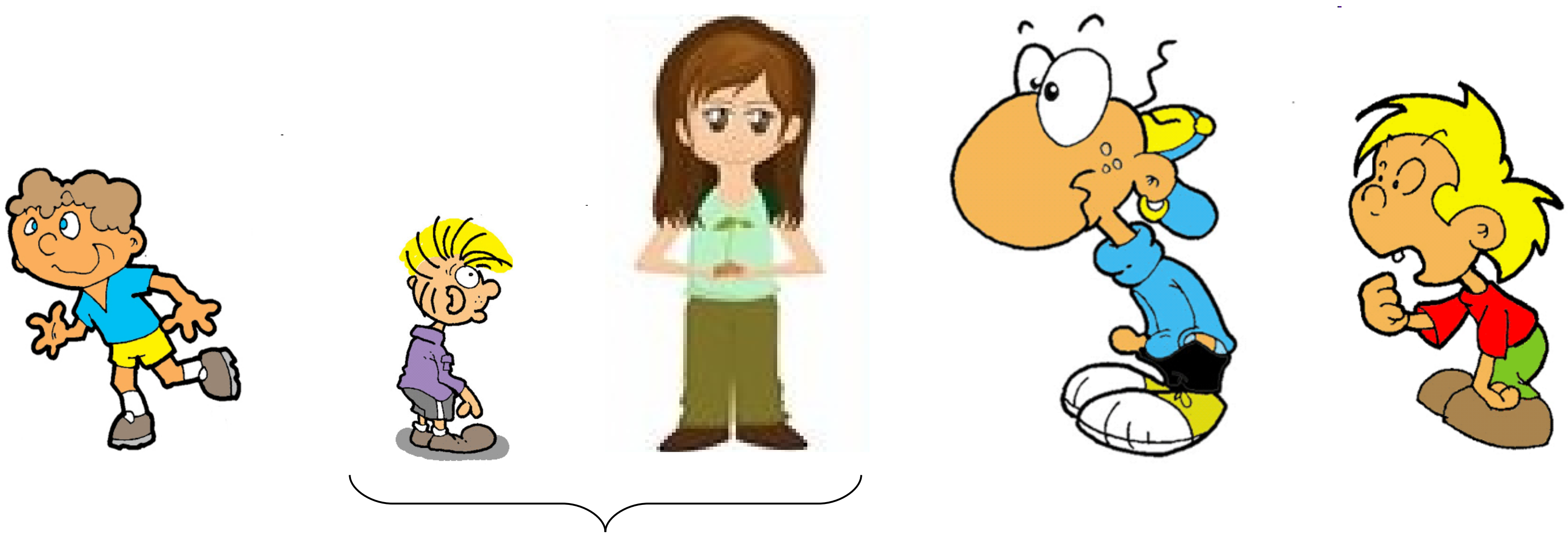
Sort Children from Shortest to Tallest



Look at NEXT two children and swap them, if necessary.

Bubble Sort

Sort Children from Shortest to Tallest

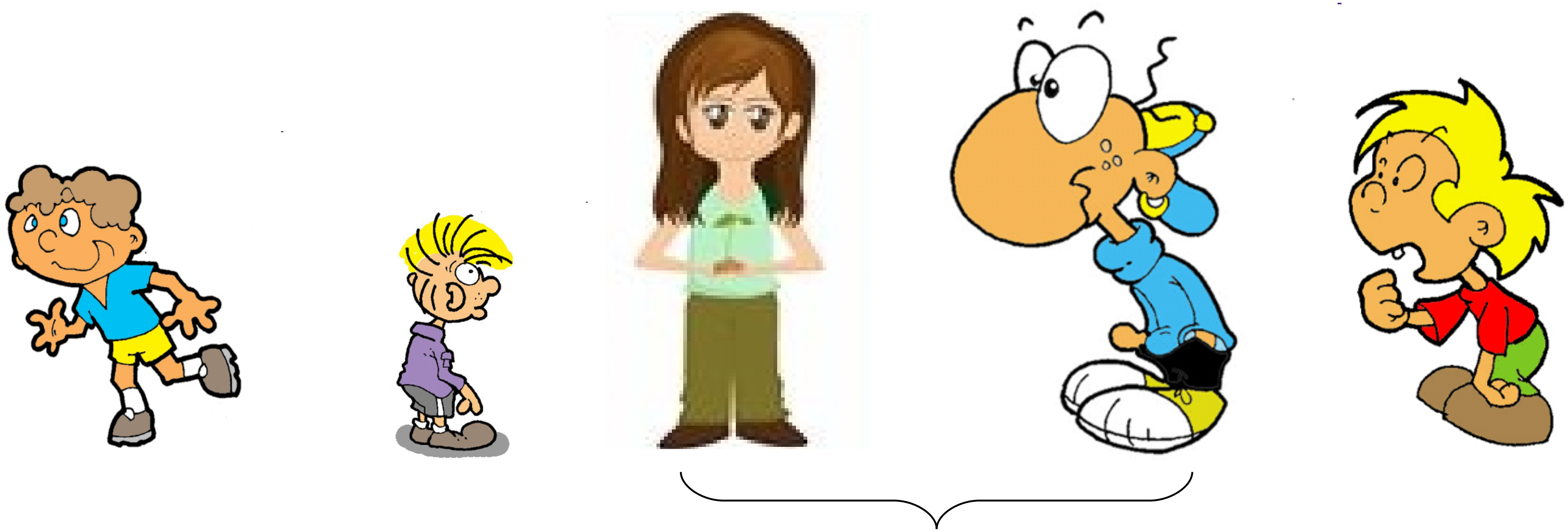


Look at NEXT two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

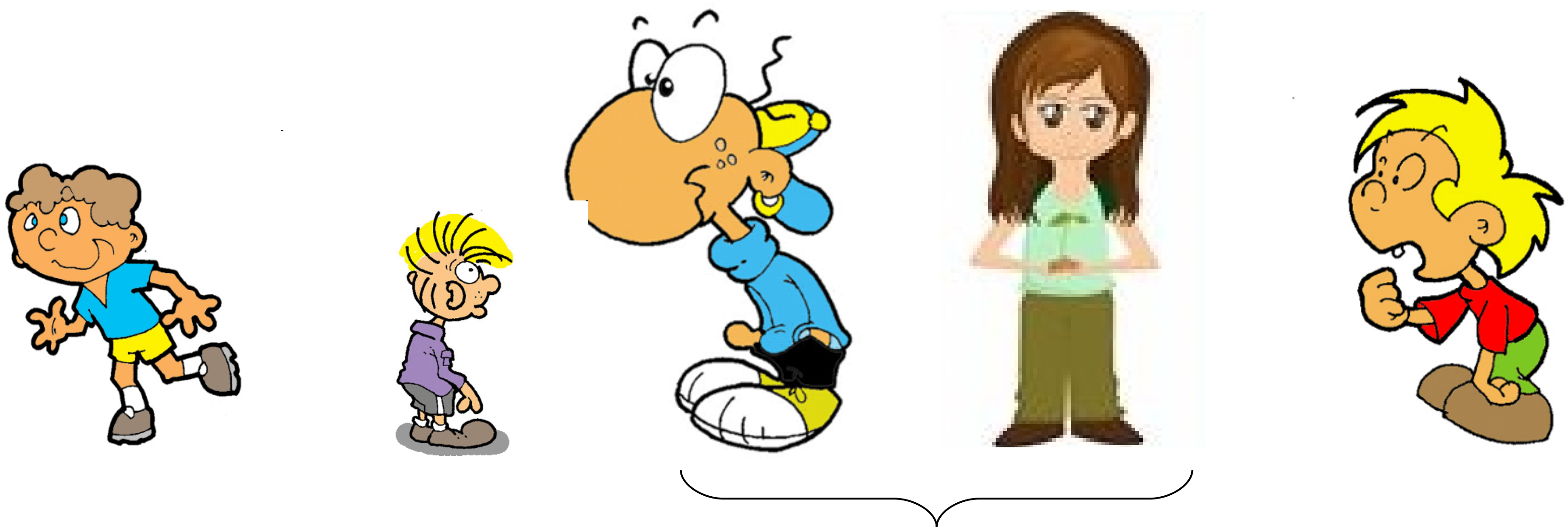


Look at NEXT two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

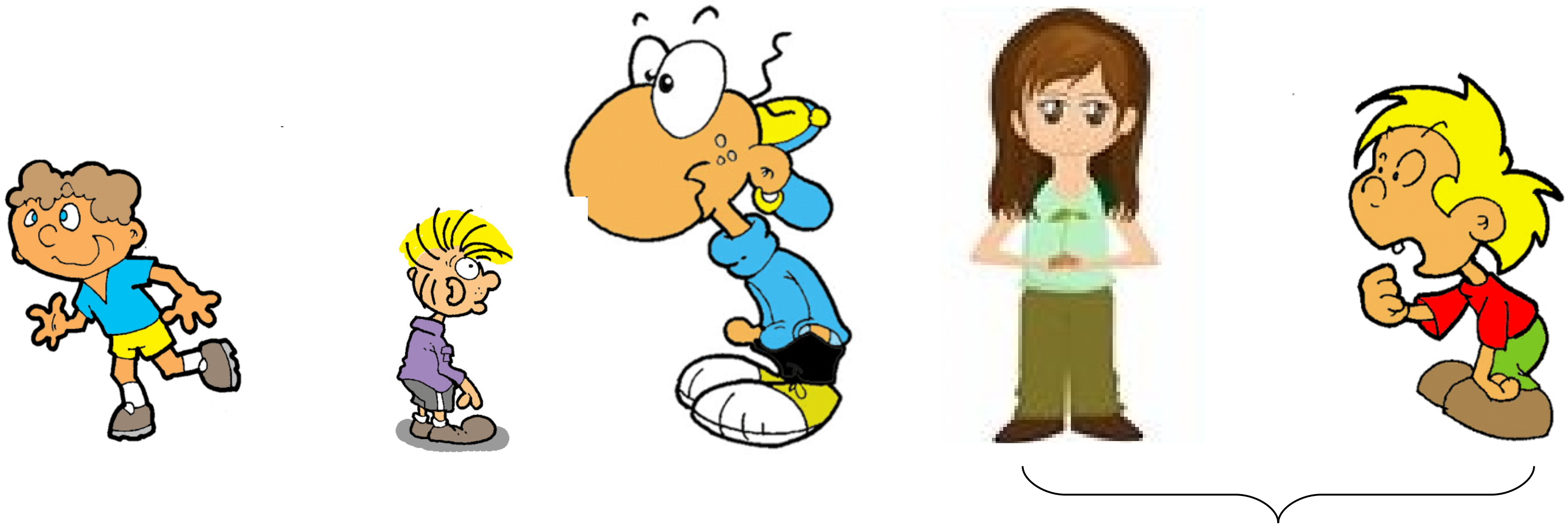


Look at NEXT two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

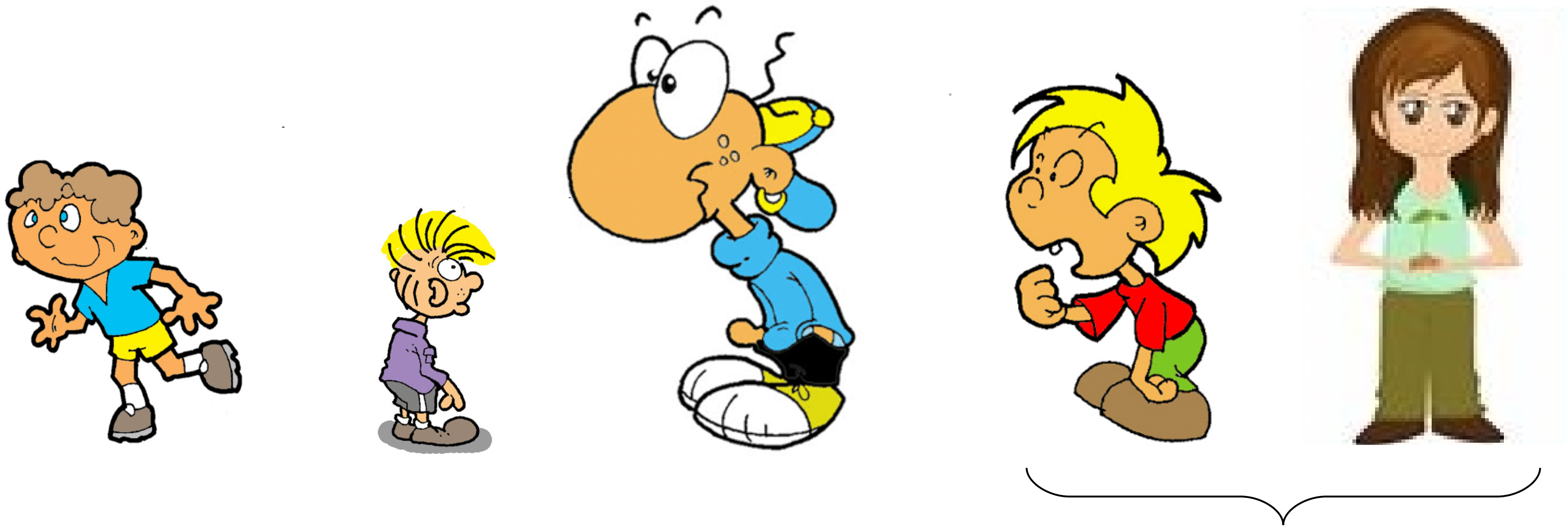


Look at LAST two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

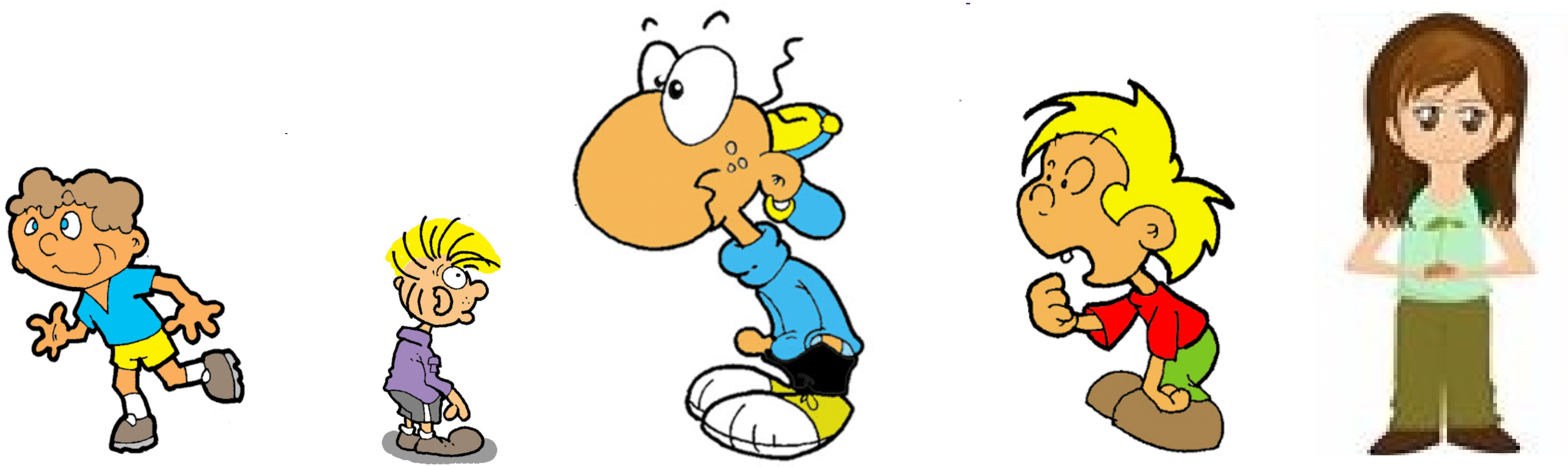
Sort Children from Shortest to Tallest



Look at LAST two children and swap them, if necessary.

Bubble Sort

Sort Children from Shortest to Tallest

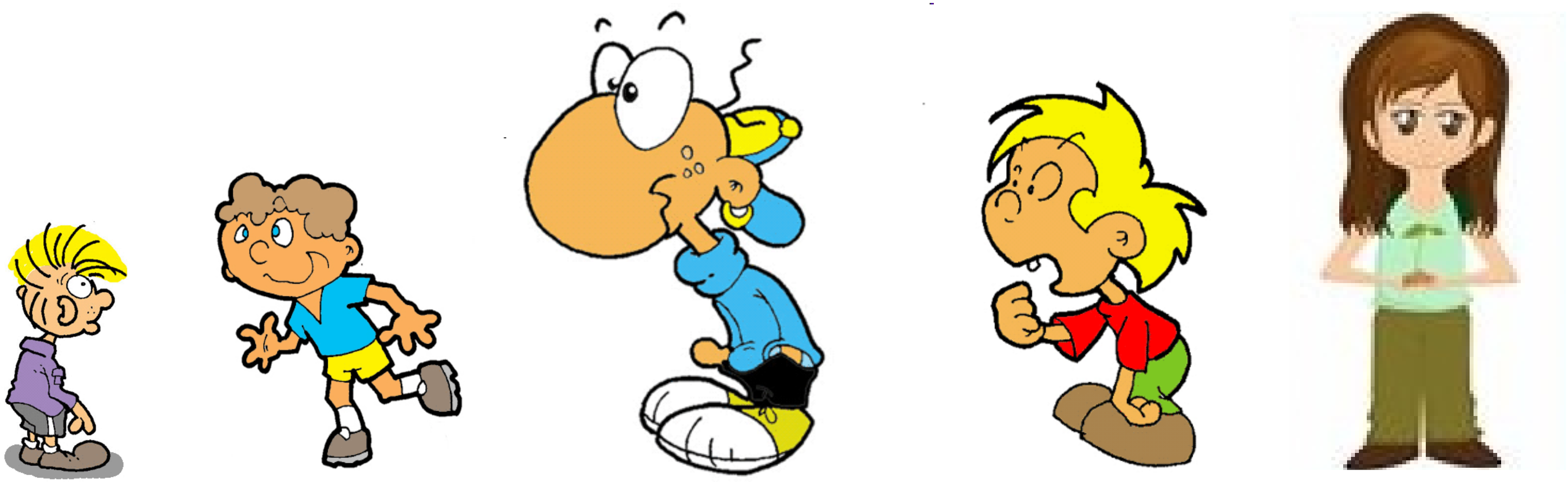


Look at FIRST two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

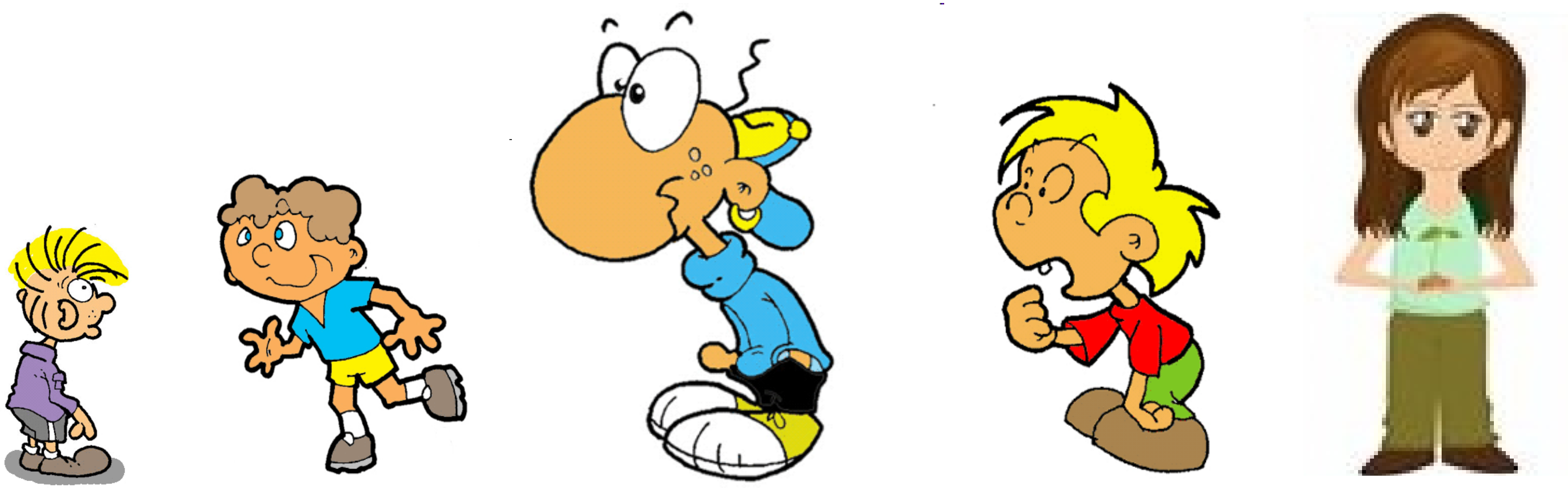


Look at FIRST two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

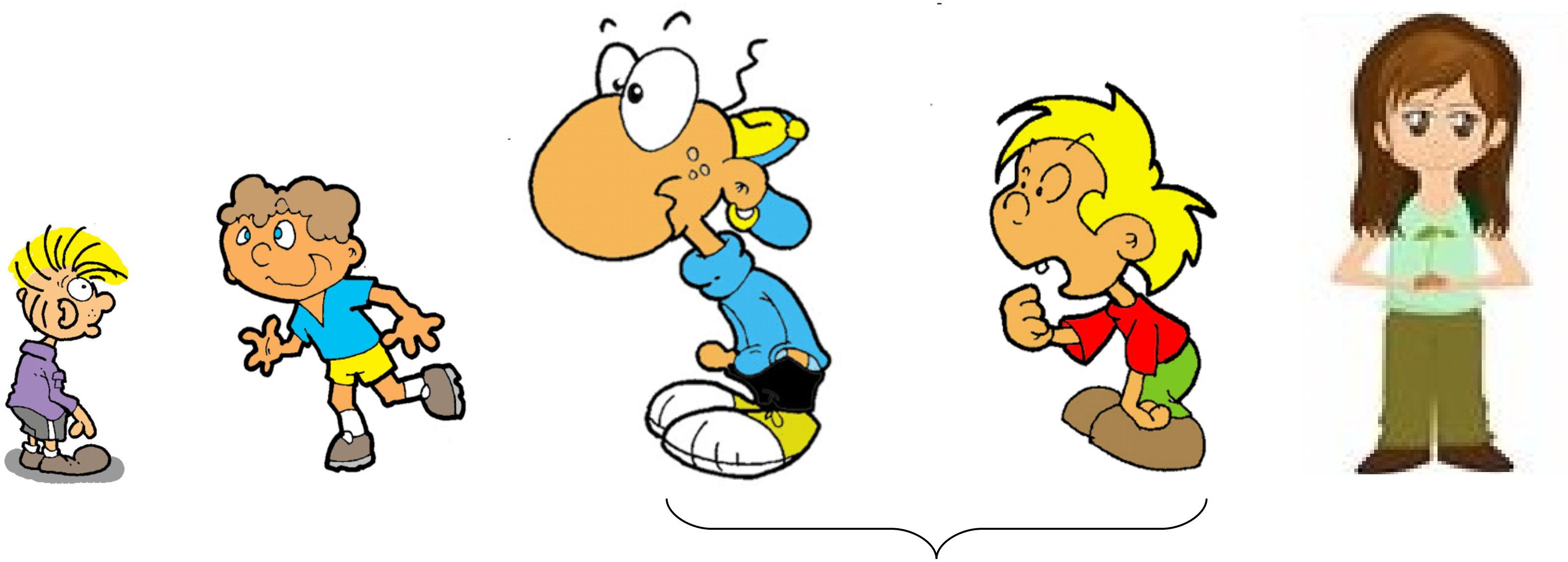


Look at NEXT two children and
swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

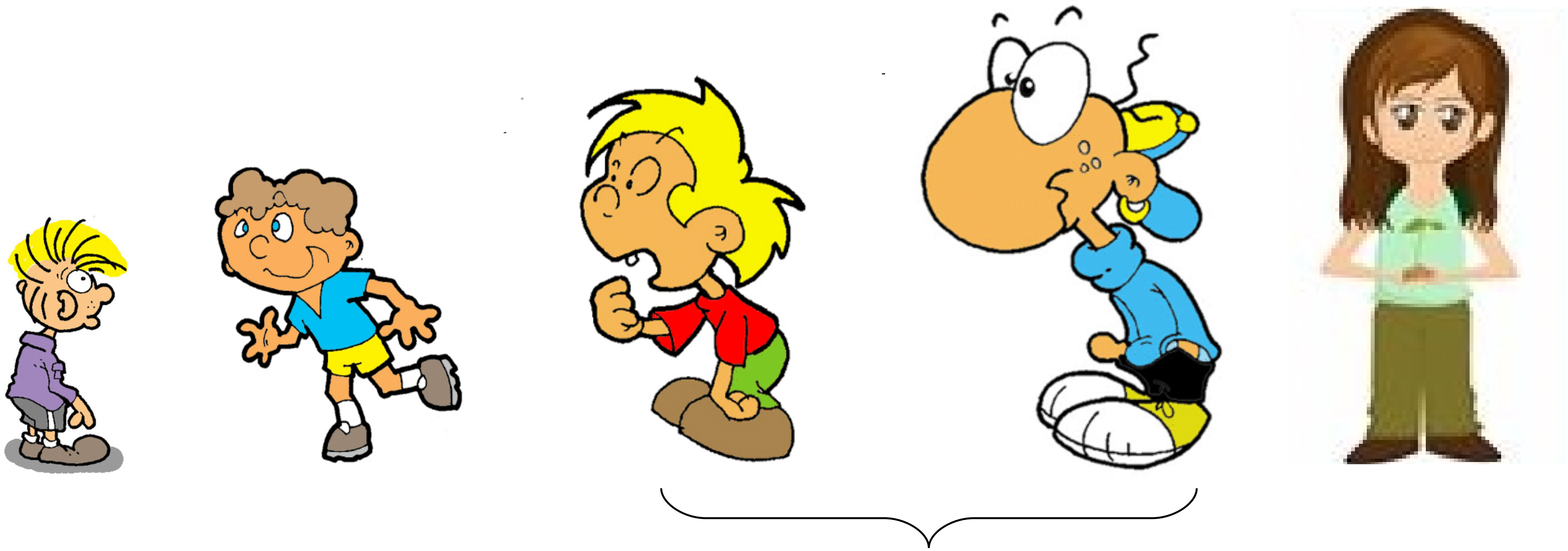


Look at NEXT two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

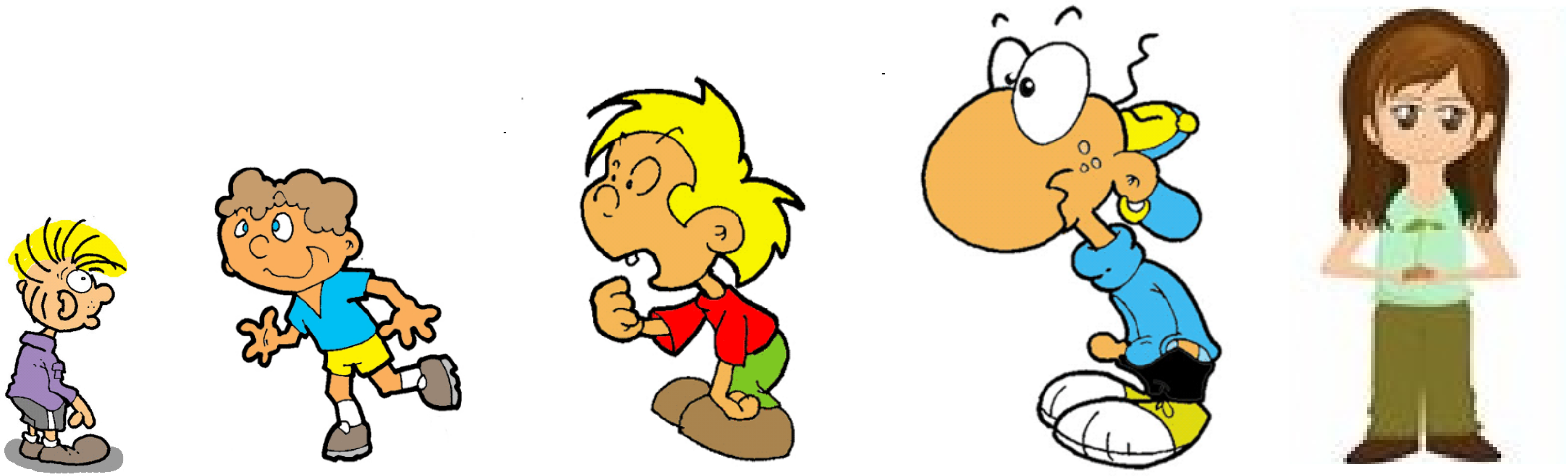
Sort Children from Shortest to Tallest



Look at NEXT two children and swap them, if necessary.

Bubble Sort

Sort Children from Shortest to Tallest

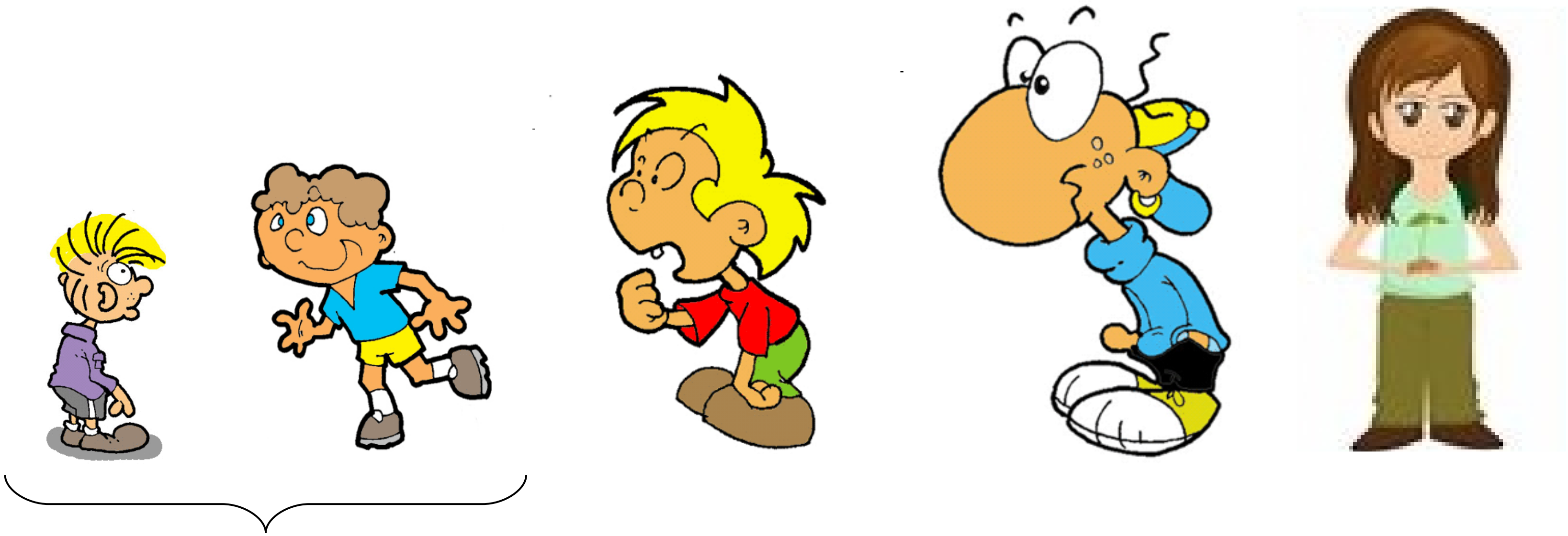


We don’t need to look at the last TWO, as in each pass the biggest “bubbles up” to the top spot.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

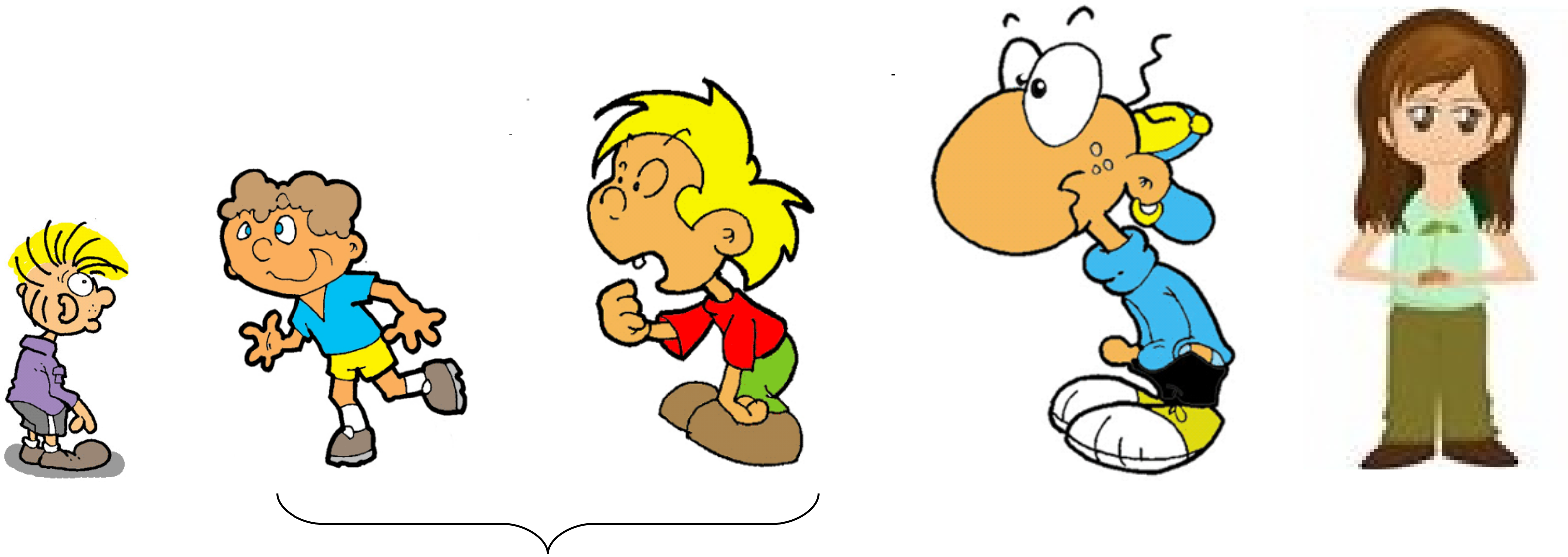


Look at FIRST two children and swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

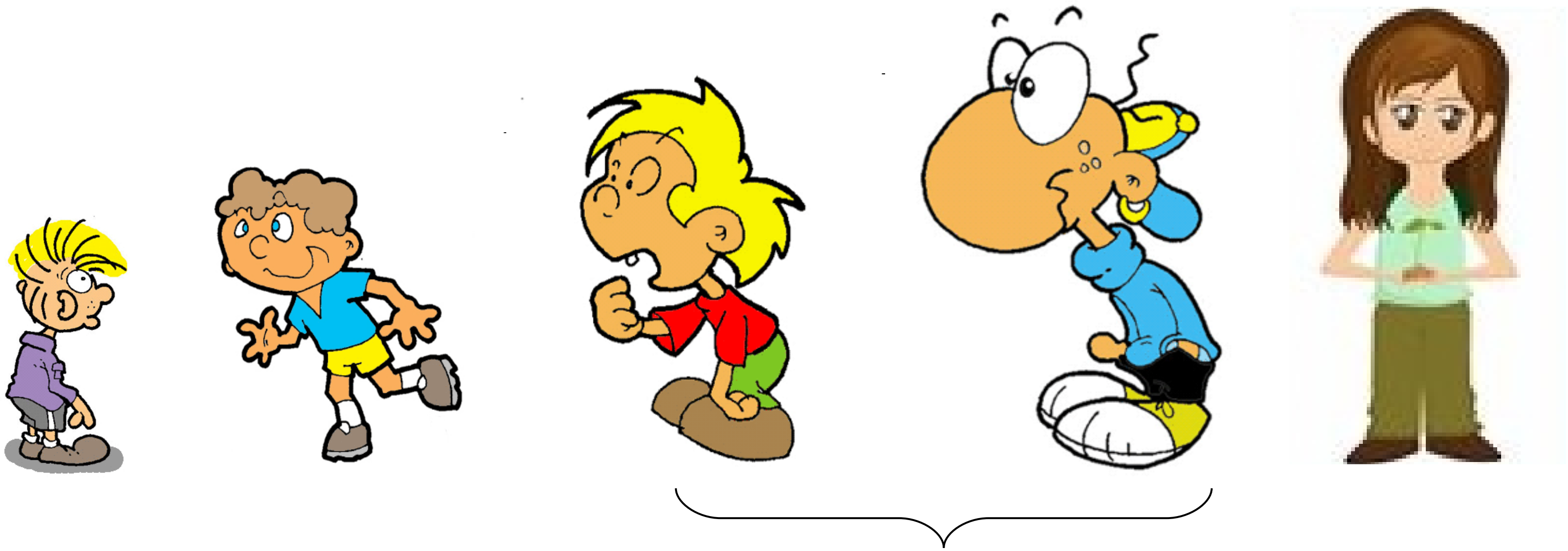


Look at NEXT two children and
swap them, if necessary.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest

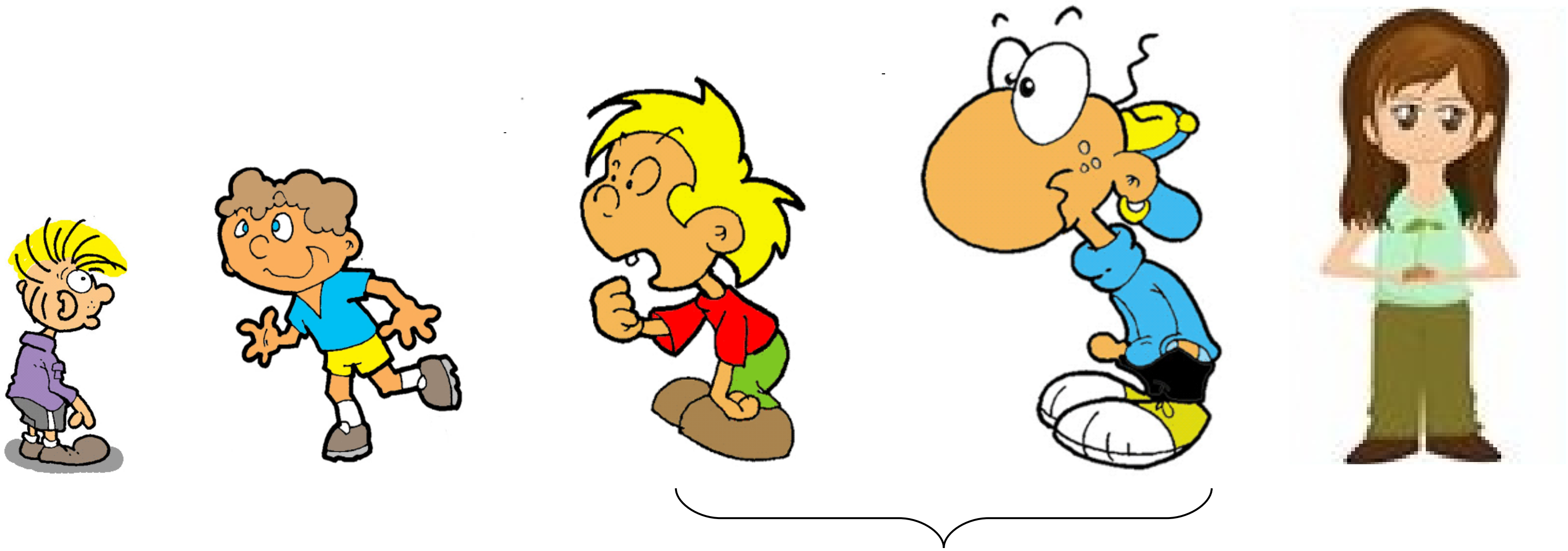


We don’t need to look at these TWO, as in each pass the biggest “bubbles up” to the top spot.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

Sort Children from Shortest to Tallest



We don't need to look at these TWO, as in each pass the biggest "bubbles up" to the top spot.

Since we didn't make ANY swaps in this pass, we're DONE!

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Bubble Sort

- How many comparisons?
- Sort 1000 items:
  - 1st pass: 999 comparisons/swaps
  - 2nd pass: 998 comparisons/swaps
  - …
  - 999th pass: 1 comparison/swap
- Sort n items:
  - (n-1)*(n-2)/2 comparison/swaps

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Insertion Sort

- Like sorting playing cards in hand:
  - Draw a first card → card is sorted.
  - Draw second card → compare with first card.
  - Draw third card → compare with two cards.
  - …
  - …
  - Draw n^{th} card → compare with n-1 cards.
- Can **speed-up** using property that hand is kept sorted.

Quicksort

- Invented in 1962 by C. Hoare.
- **Much** more efficient than previous algorithms.
  - Fewer steps needed to get items into order.
- Still widely used today.
- Uses **recursion**.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Recursion

Contains smaller version of same picture

Recursion in Computing

- Task is broken down into smaller/easier tasks solved in a similar way.
  - Ends when we hit a BASE/END case.

Recursion in Computing

- Task is broken down into smaller/easier tasks solved in a similar way.
  - Ends when we hit a BASE/END case.
  - Invest $1000 at 8% interest.
  - How much do I have in 3 years?
    - Cash(3 years) = Cash(2 years) + 8% x Cash(2 years)
    - Cash(2 years) = Cash(1 years) + 8% x Cash(1 years)
    - Cash(1 year) = Cash(0 years) + 8% x Cash(0 years)
    - Cash(0 years) = $1000.00

    - Cash(3 years) = $1259.71!

Quicksort Core Action

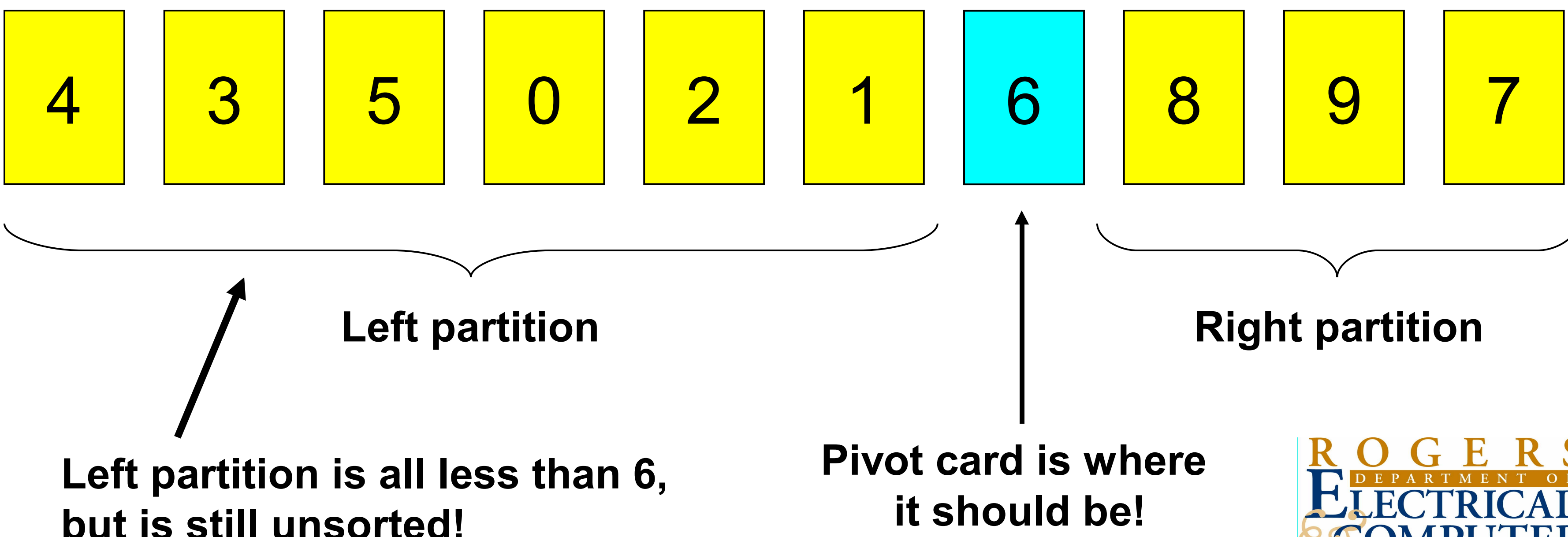
- Want to sort 10 cards:

6 3 9 0 2 1 7 4 5 8

- Choose a "pivot" card: 6

Quicksort Core Action

- Walk along the cards and "**partition**" the cards into two groups:
  - Cards smaller than the pivot.
  - Cards larger than the pivot.



Quicksort Core Action

- Now take the left partition & repeat action!

4 3 5 0 2 1

- Choose a "pivot" card: 4

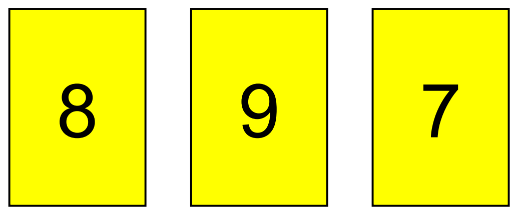
Pivot is in its correct place

- Partition the left partition:

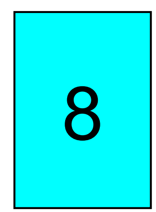
2 3 1 0 4 5

Quicksort Core Action

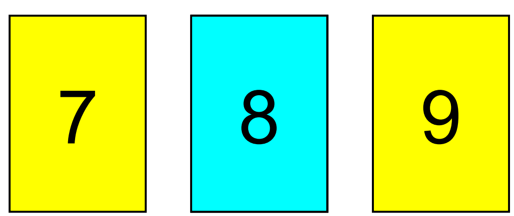
- Likewise, take the right partition and repeat the process.



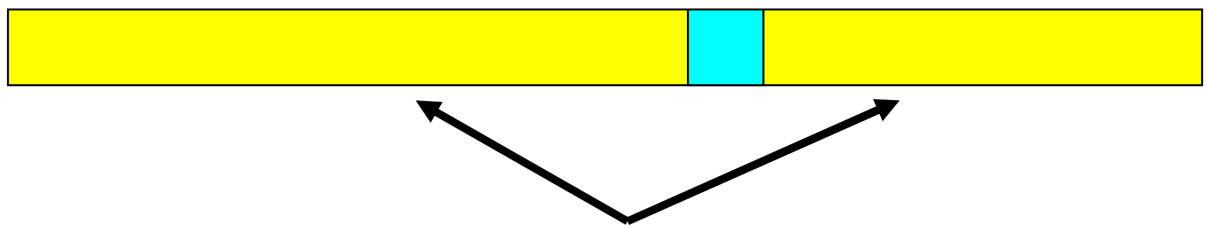
- Choose a “pivot” card:



- Partition the right partition:

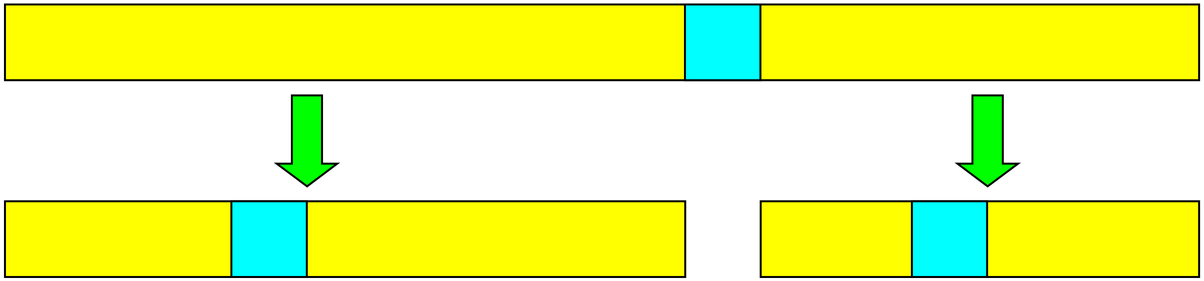


Quicksort Overview

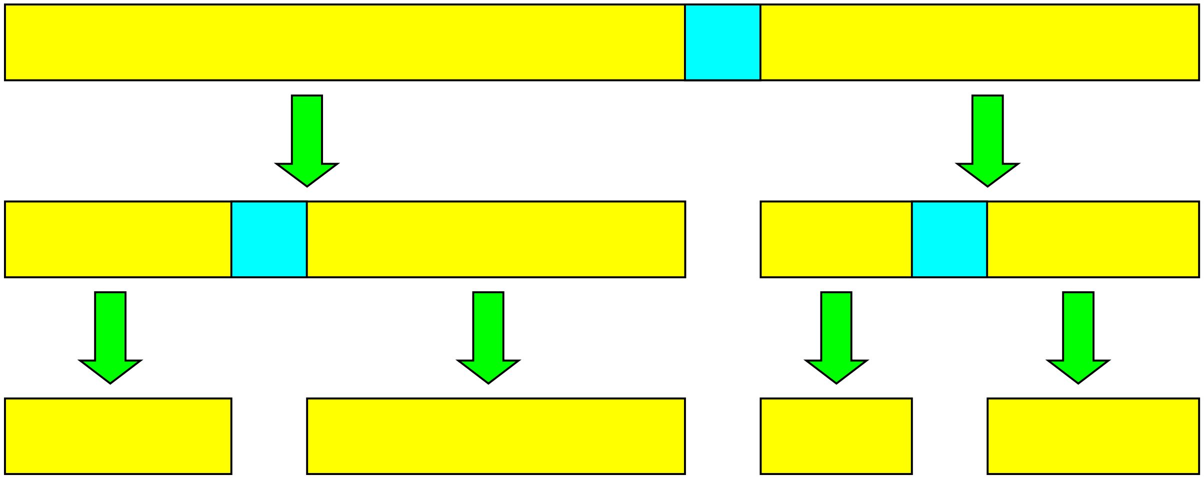


Pivot and partition the left and right half.

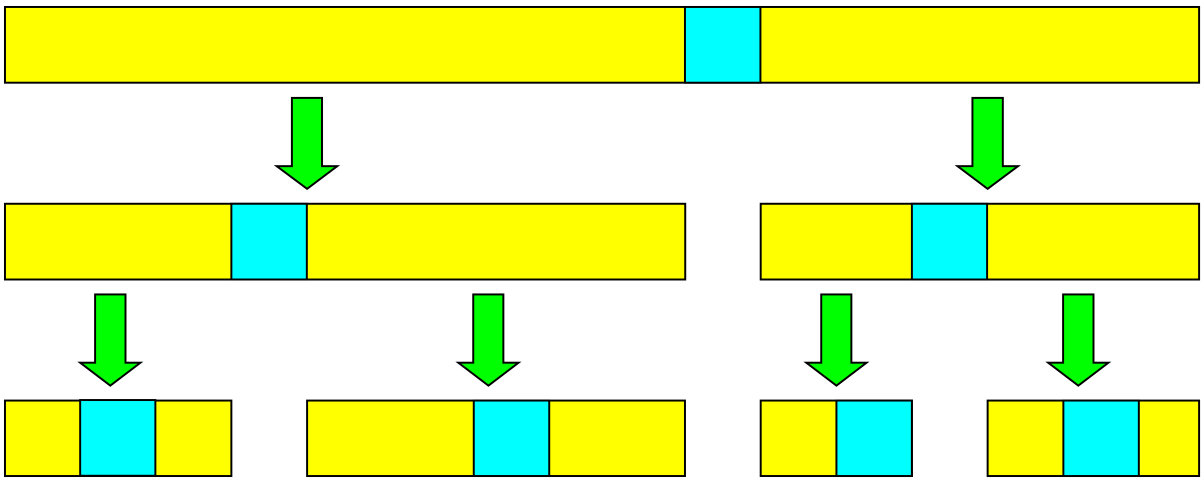
Quicksort Overview



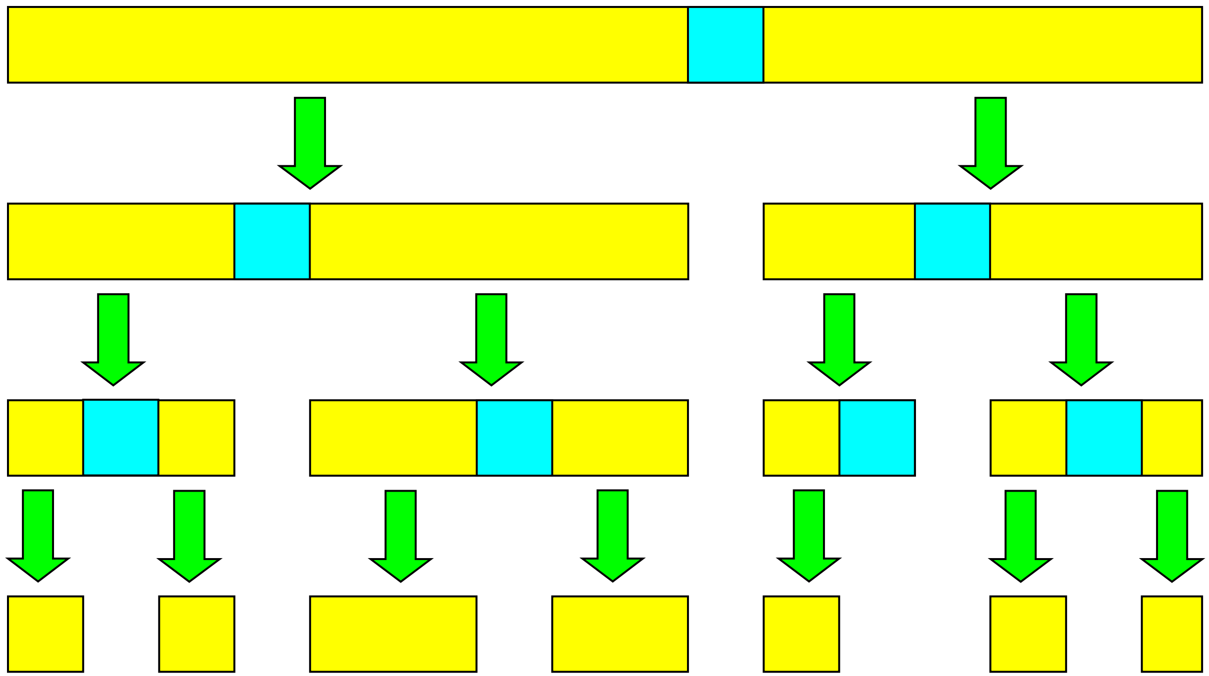
Quicksort Overview



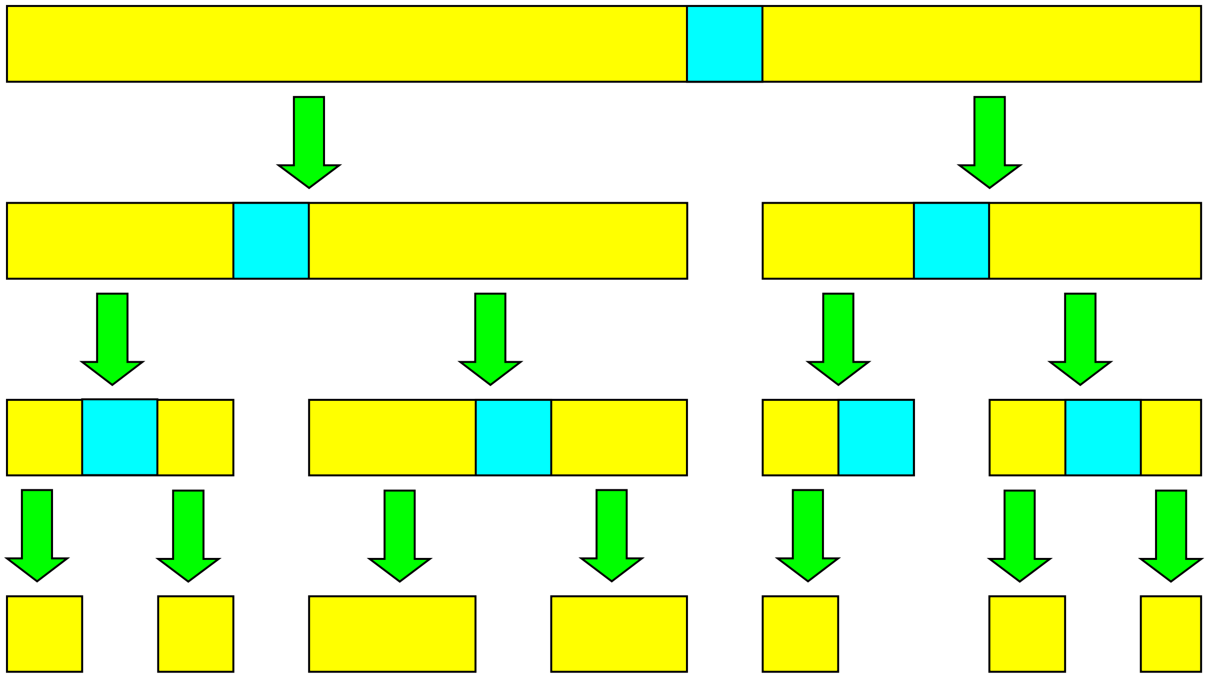
Quicksort Overview



Quicksort Overview



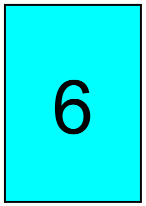
Quicksort Overview

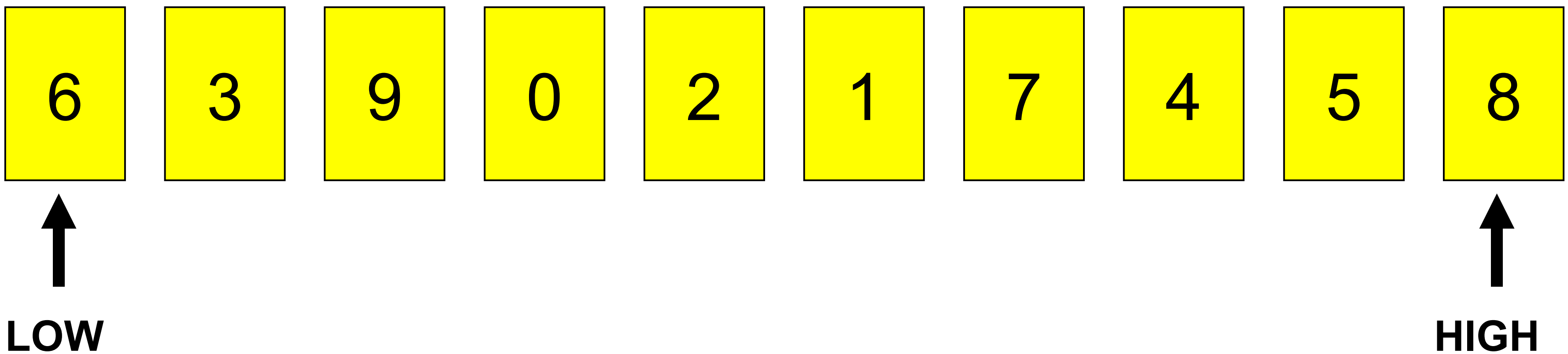


Continue until the pieces are so small there is nothing to do! $\rightarrow$ SORTED

Nitty Gritty Details

- How do we do the partitioning?

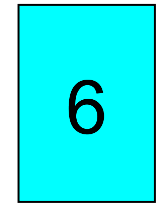


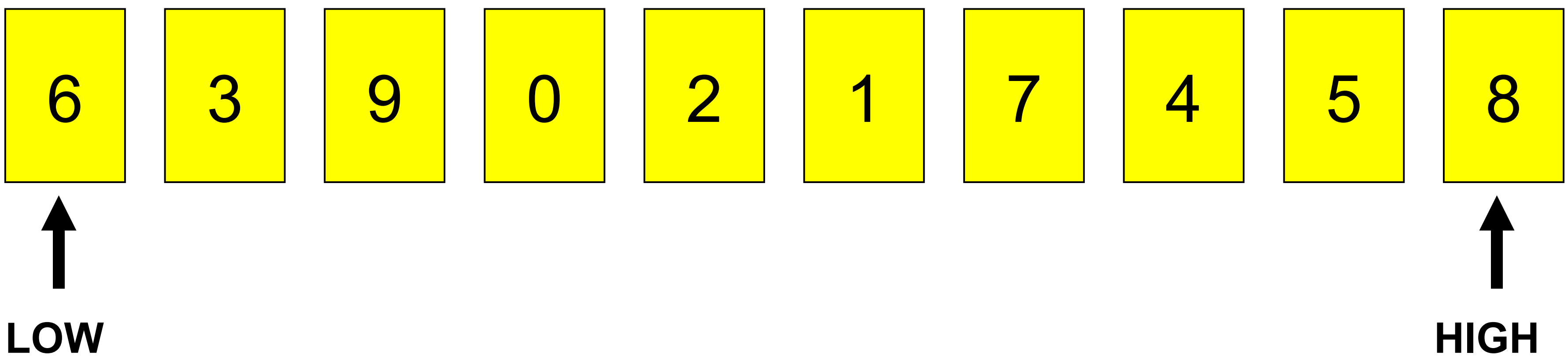


- Consider two “arrows” LOW and HIGH.

Nitty Gritty Details

- How do we do the partitioning?

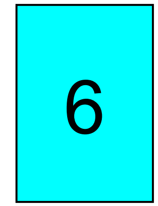


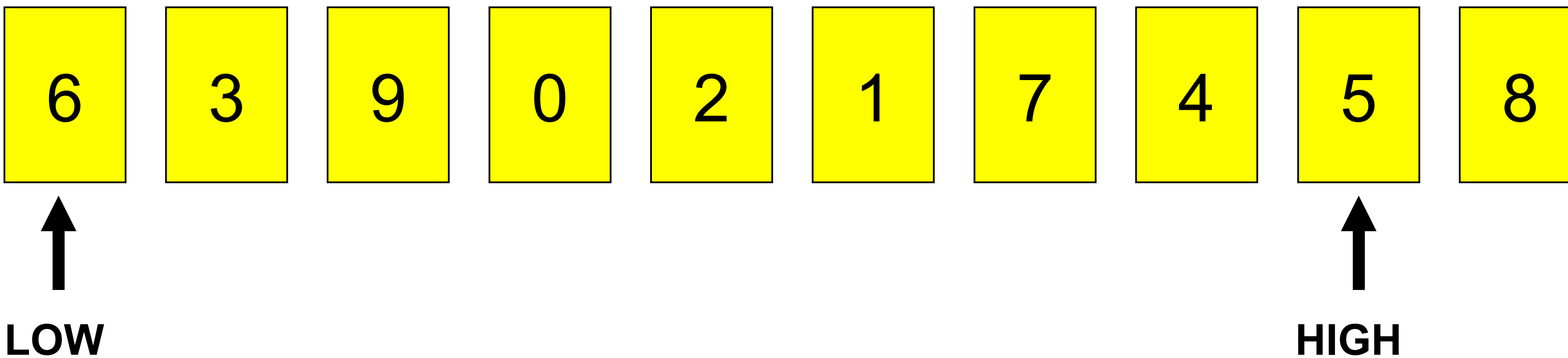


- Consider two “arrows” LOW and HIGH.
- HIGH will march LEFT until it finds a number less than PIVOT.

Nitty Gritty Details

- How do we do the partitioning?

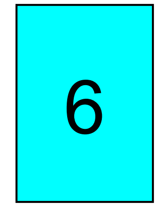


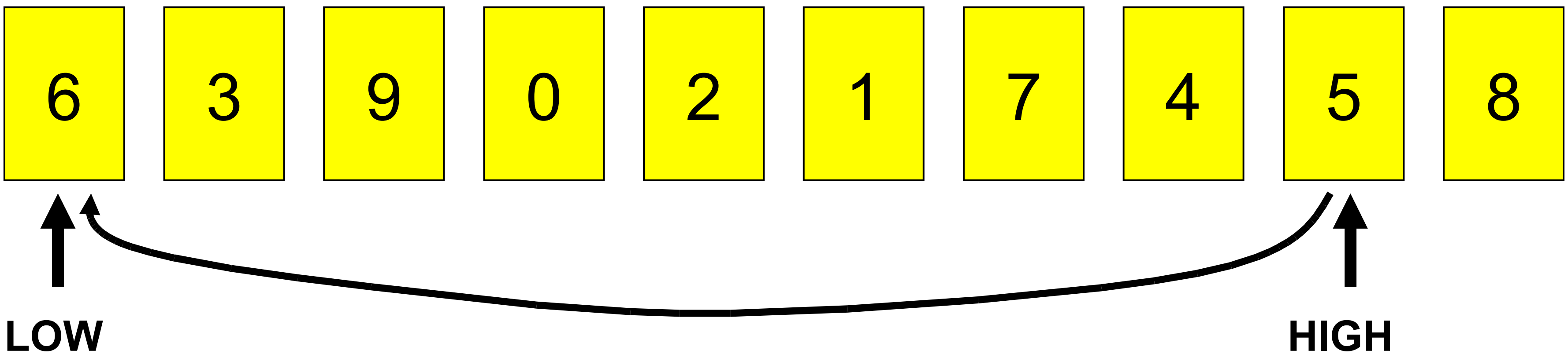


- Consider two “arrows” LOW and HIGH.
- HIGH marches LEFT until it finds a number less than PIVOT.

Nitty Gritty Details

- How do we do the partitioning?



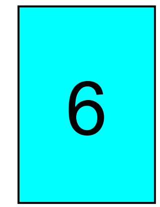


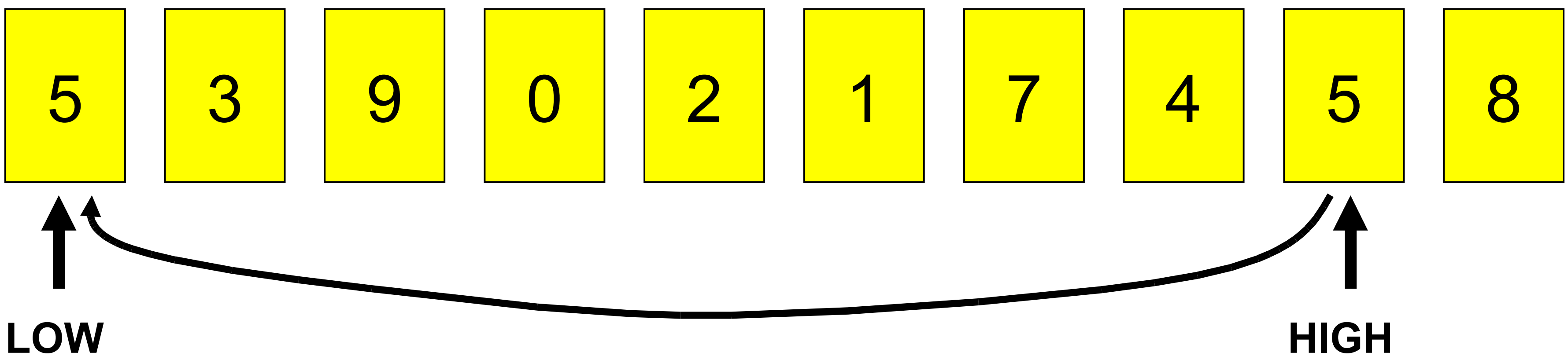
- The number at HIGH is moved to the position LOW.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

- How do we do the partitioning?

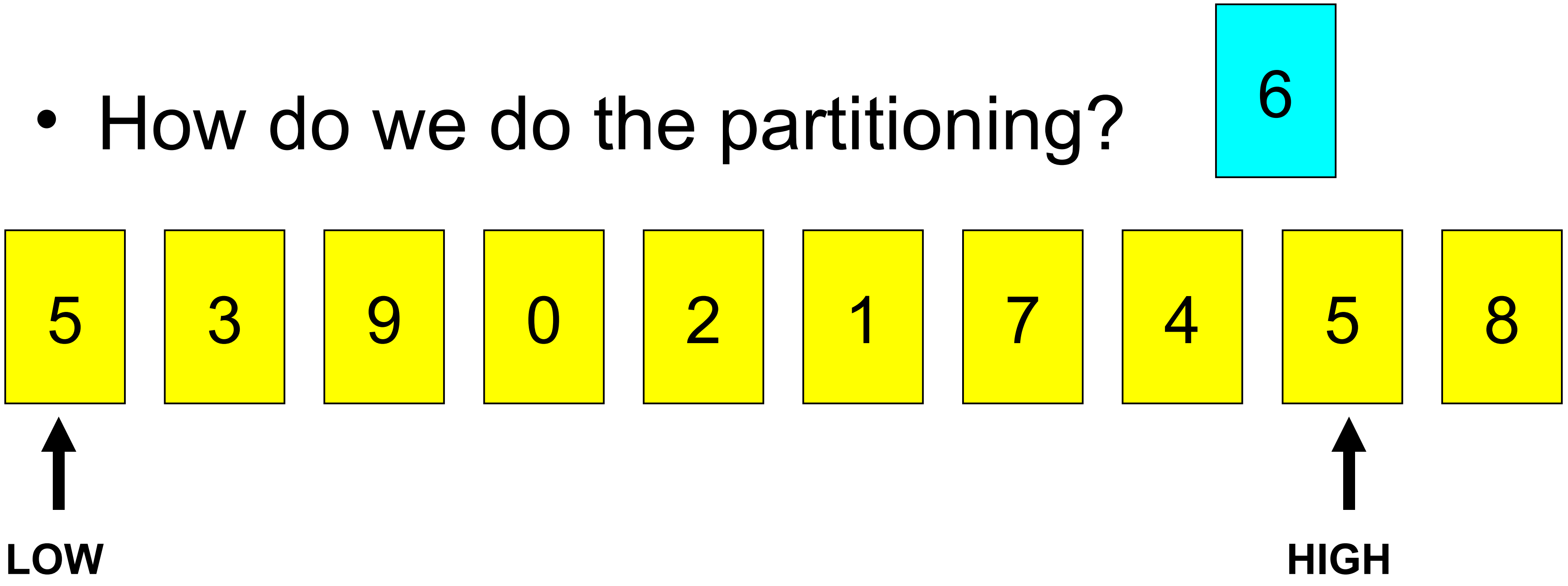




- The number at HIGH is moved to the position LOW.

Nitty Gritty Details

- How do we do the partitioning?

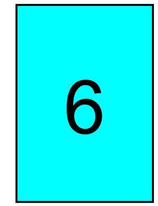


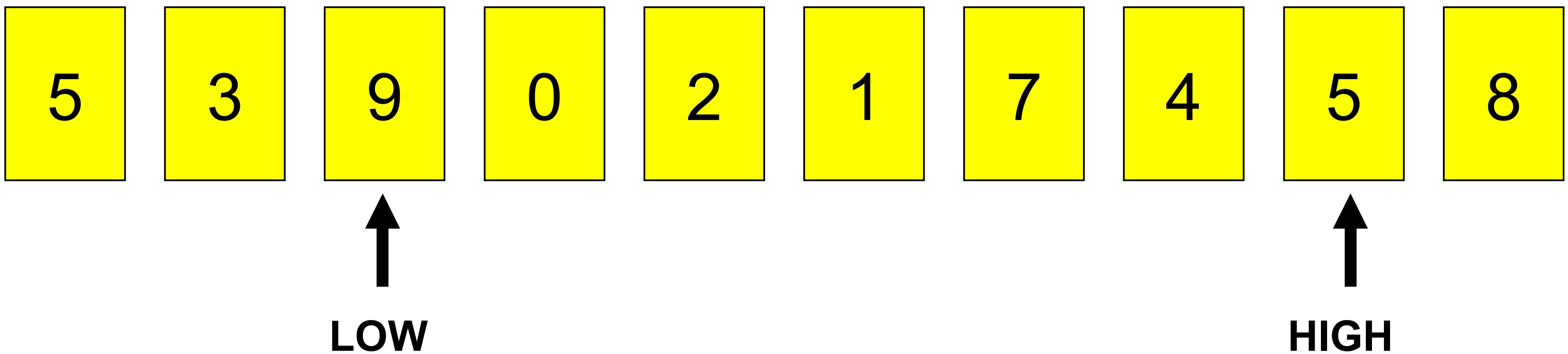
- LOW marches RIGHT until it finds a number greater than PIVOT

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

- How do we do the partitioning?



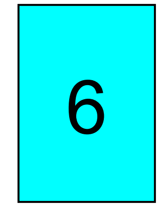


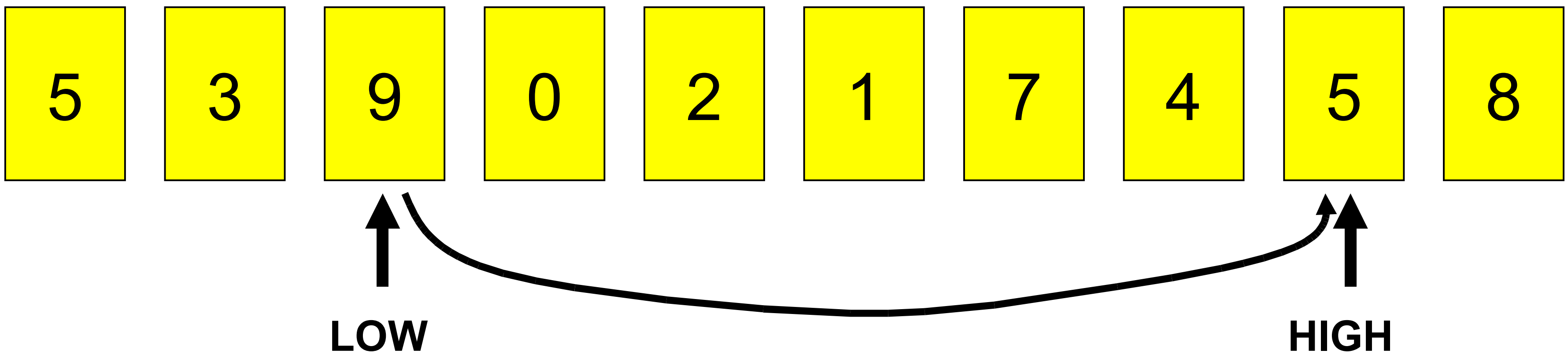
- LOW marches RIGHT until it finds a number greater than PIVOT

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

- How do we do the partitioning?

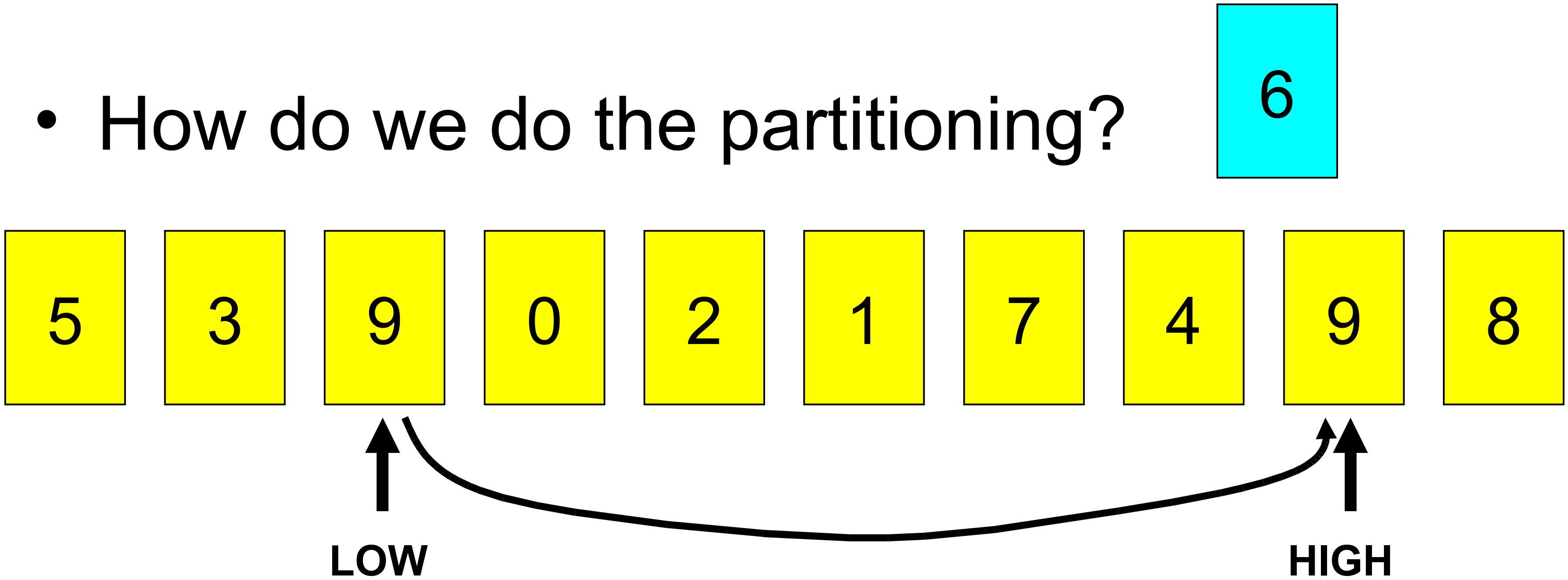




- The number at position LOW is moved to position HIGH.

Nitty Gritty Details

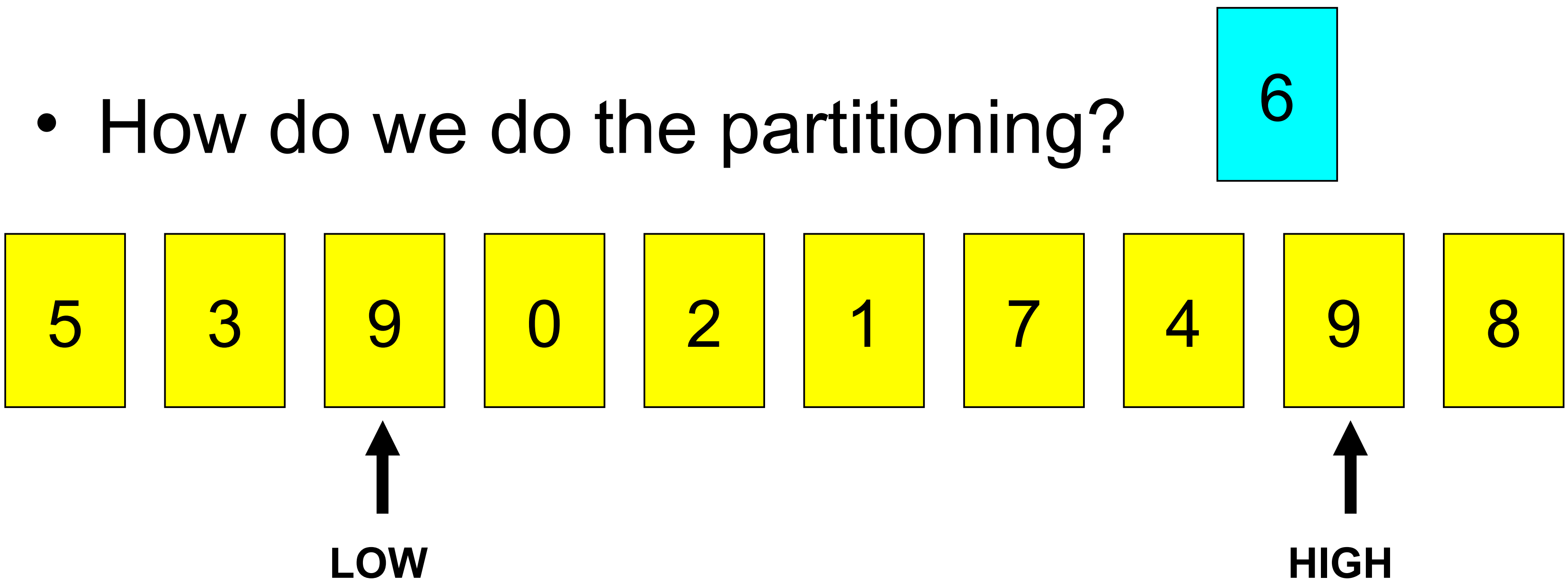
- How do we do the partitioning?



- The number at position LOW is moved to position HIGH.

Nitty Gritty Details

- How do we do the partitioning?

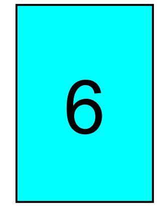


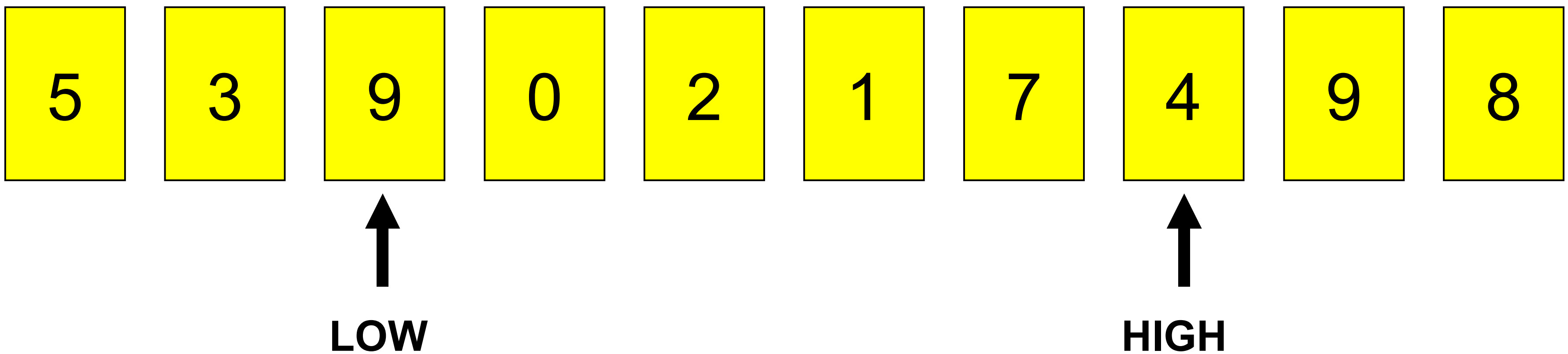
- HIGH marches left until it finds a number less than PIVOT.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

- How do we do the partitioning?



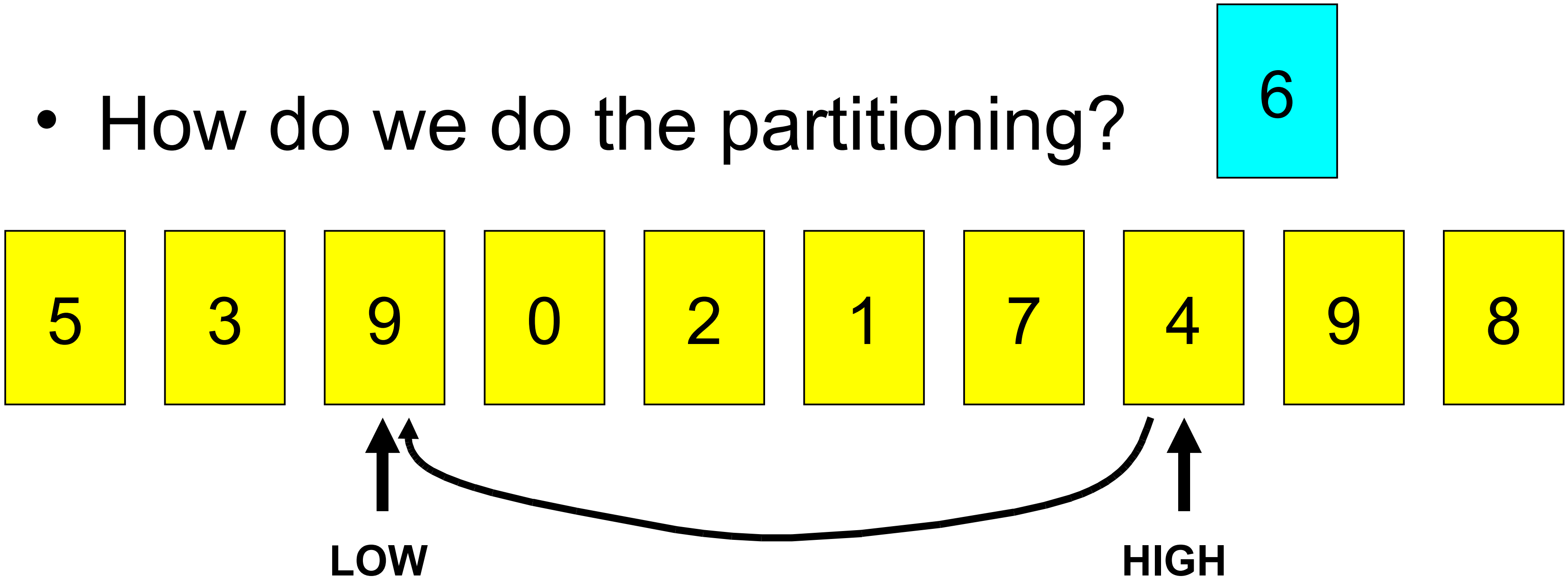


- HIGH marches left until it finds a number less than PIVOT.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

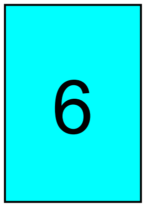
- How do we do the partitioning?

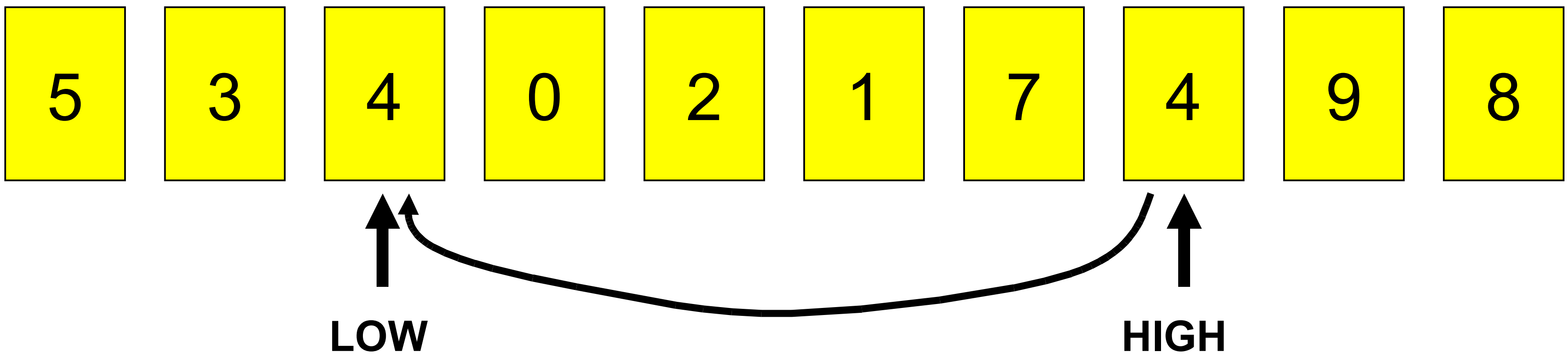


- The number at position HIGH is moved to position LOW.

Nitty Gritty Details

- How do we do the partitioning?

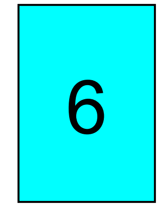


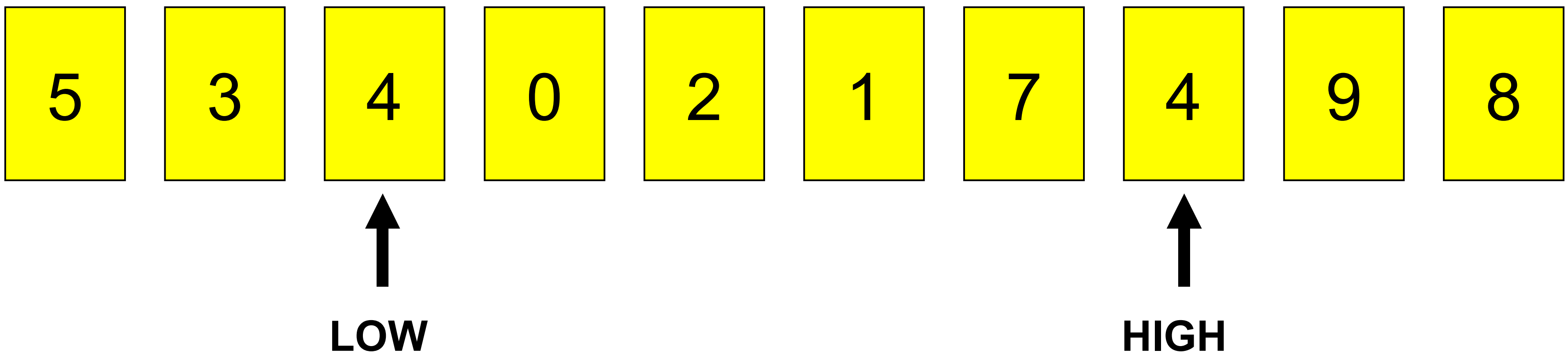


- The number at position HIGH is moved to position LOW.

Nitty Gritty Details

- How do we do the partitioning?



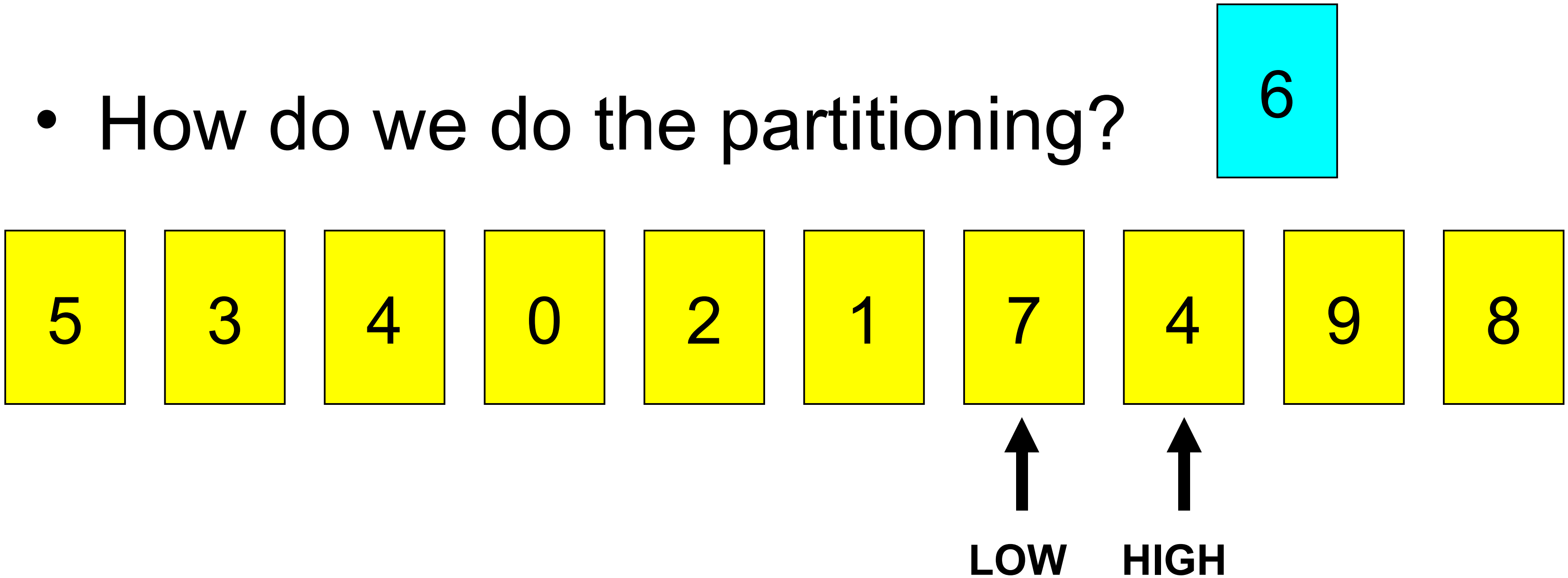


- LOW marches right until it finds a number larger than PIVOT.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

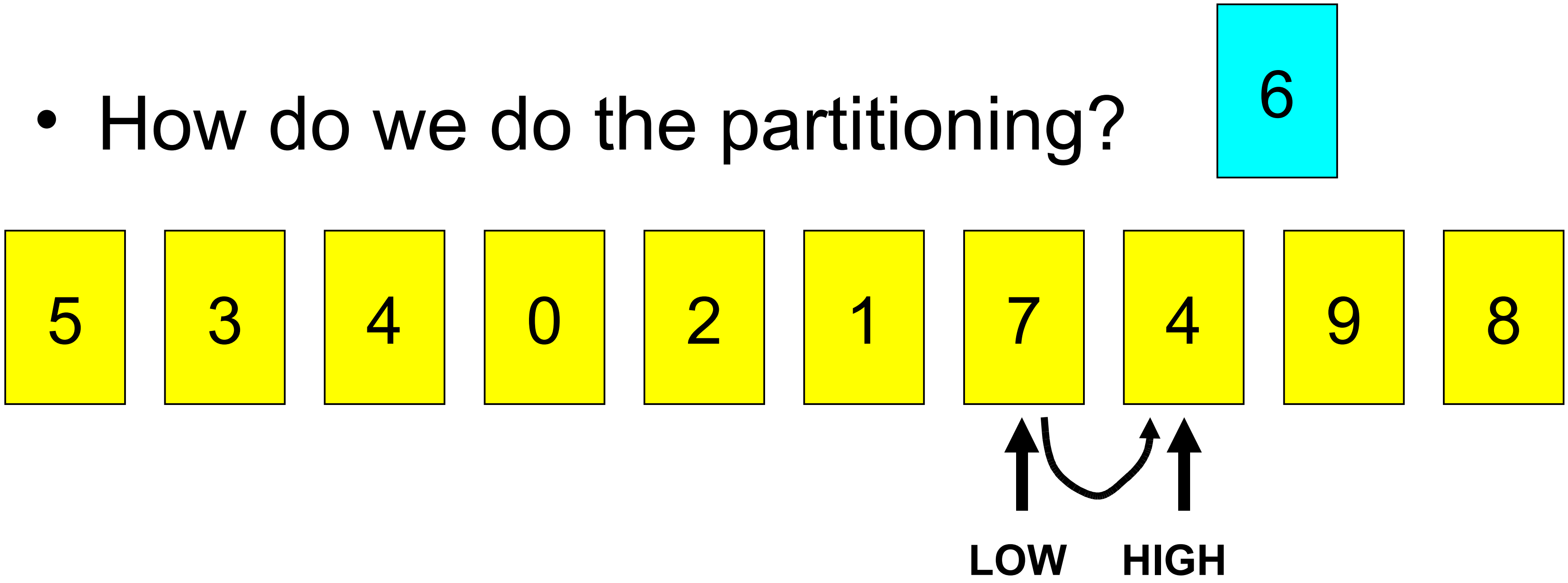
- How do we do the partitioning?



- LOW marches right until it finds a number larger than PIVOT.

Nitty Gritty Details

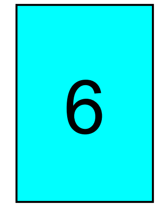
- How do we do the partitioning?

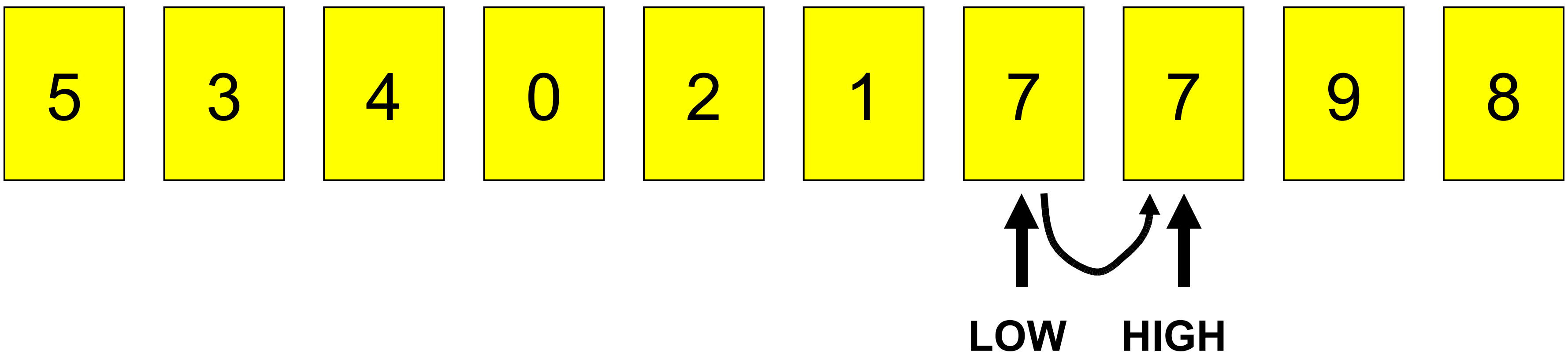


- The number at position LOW is moved to position HIGH.

Nitty Gritty Details

- How do we do the partitioning?



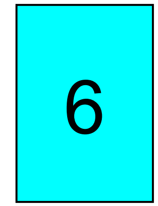


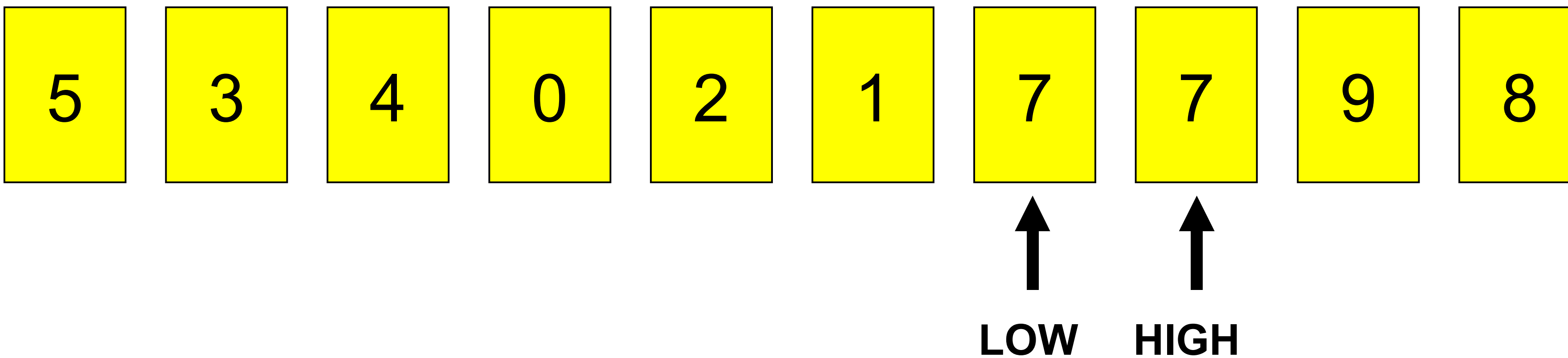
- The number at position LOW is moved to position HIGH.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

- How do we do the partitioning?

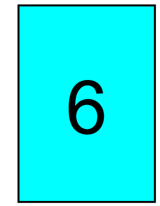


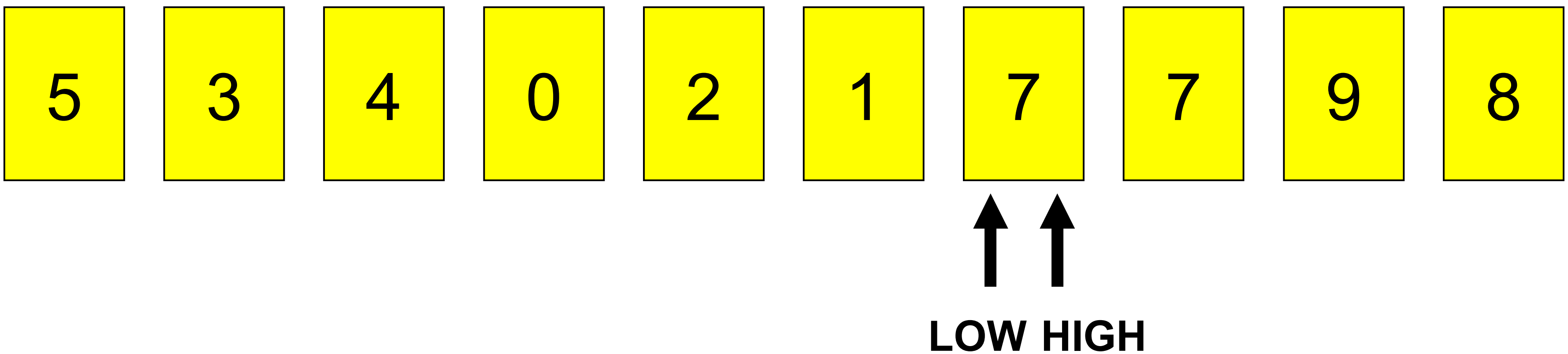


- HIGH marches LEFT until it aligns with LOW.

Nitty Gritty Details

- How do we do the partitioning?

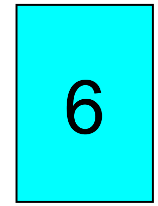


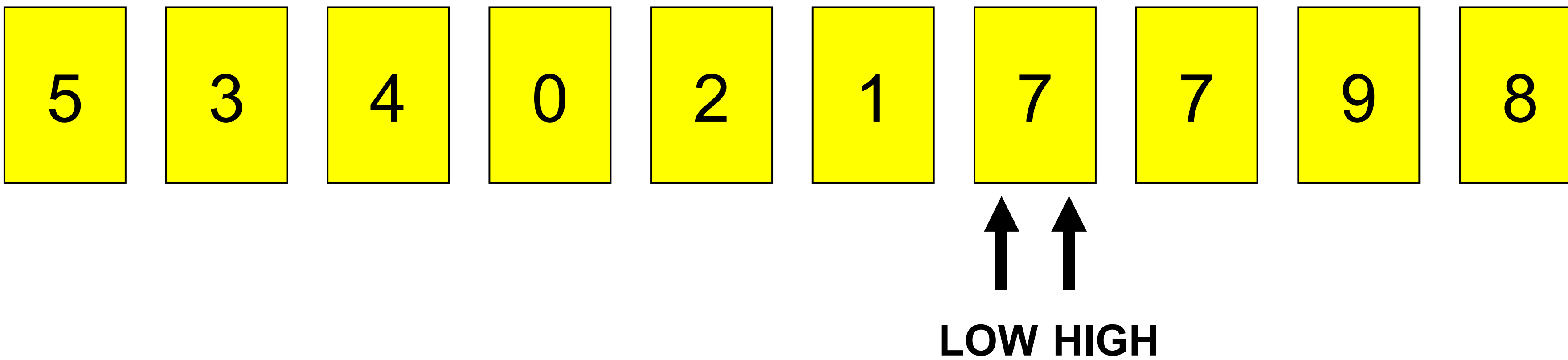


- HIGH marches LEFT until it aligns with LOW.

Nitty Gritty Details

- How do we do the partitioning?

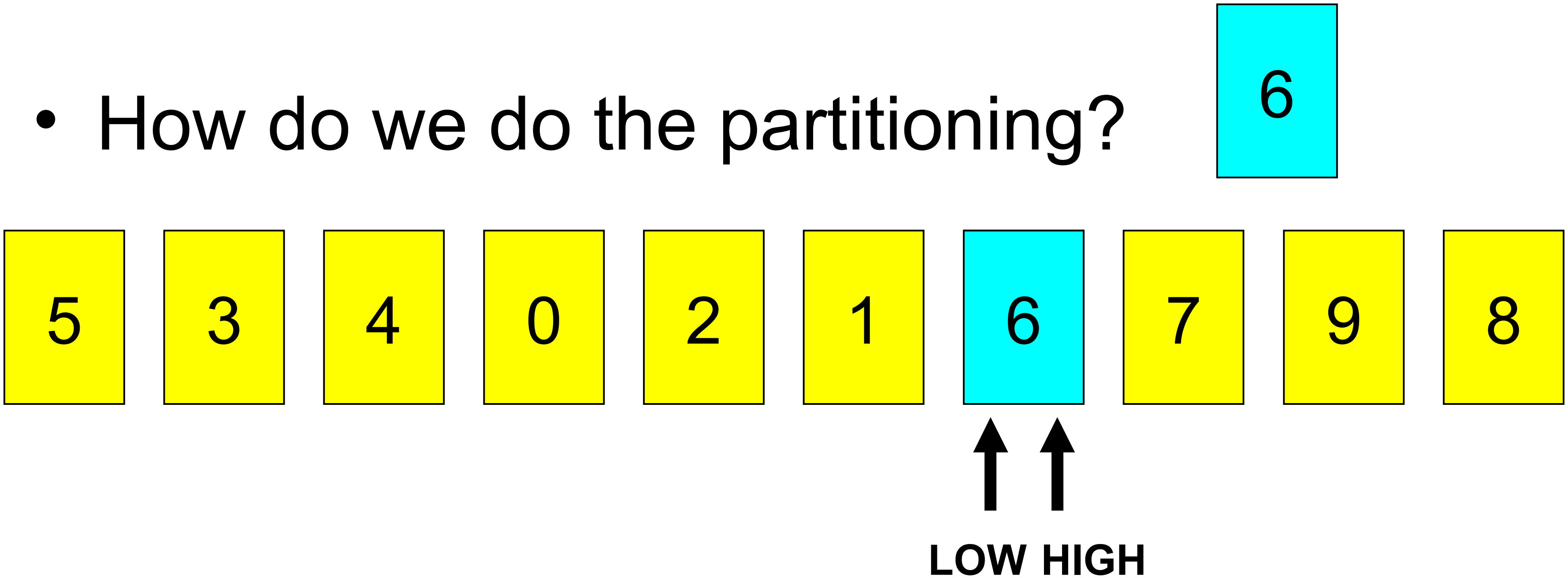




- The PIVOT is inserted at the position LOW/HIGH.

Nitty Gritty Details

- How do we do the partitioning?

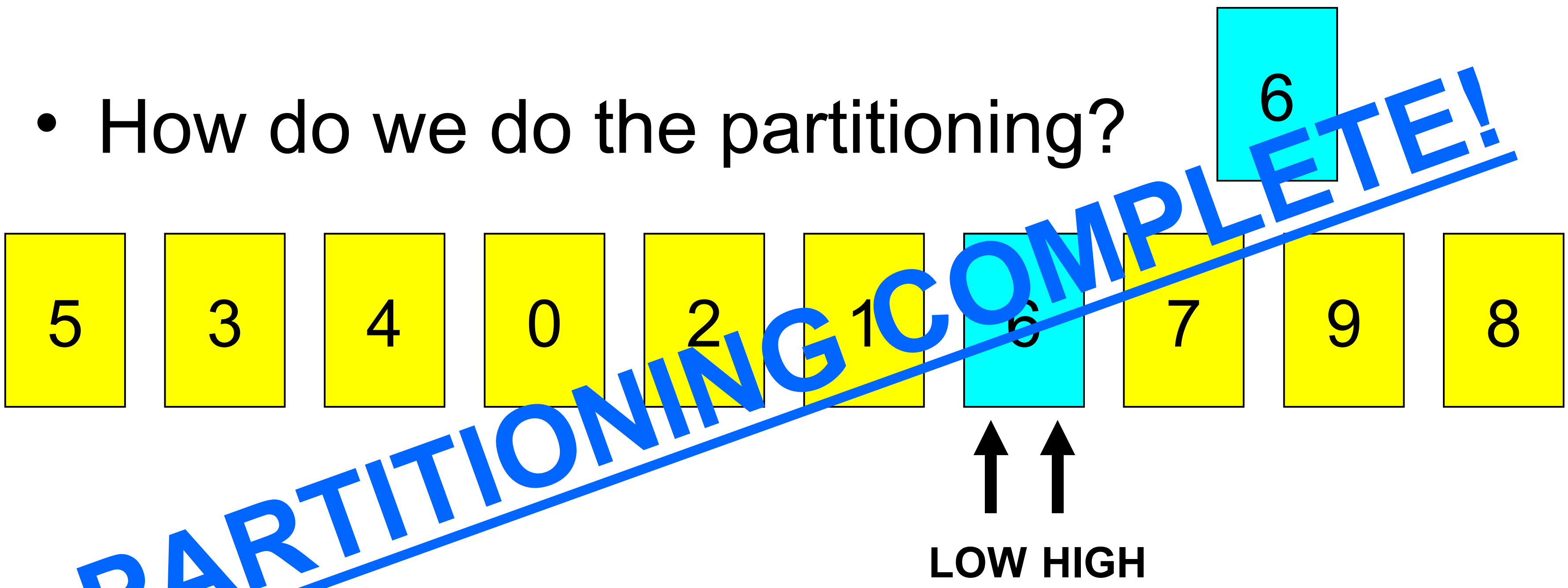


- The PIVOT is inserted at the position LOW/HIGH.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

Nitty Gritty Details

- How do we do the partitioning?



- The PIVOT is inserted at the position LOW/HIGH.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO

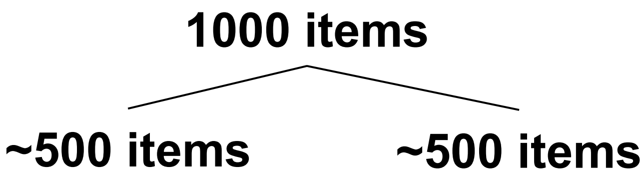
Quicksort

- How many comparisons/swaps?
  - Sort 1000 items:

1000 items

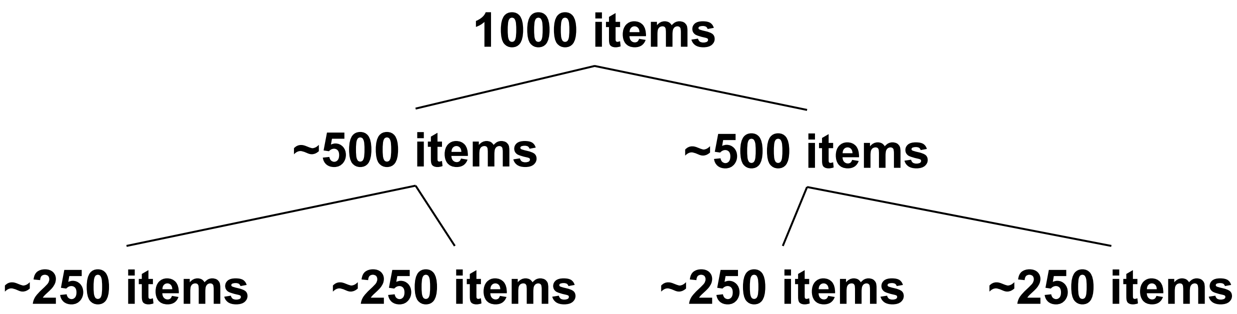
Quicksort

- How many comparisons/swaps?
  - Sort 1000 items:



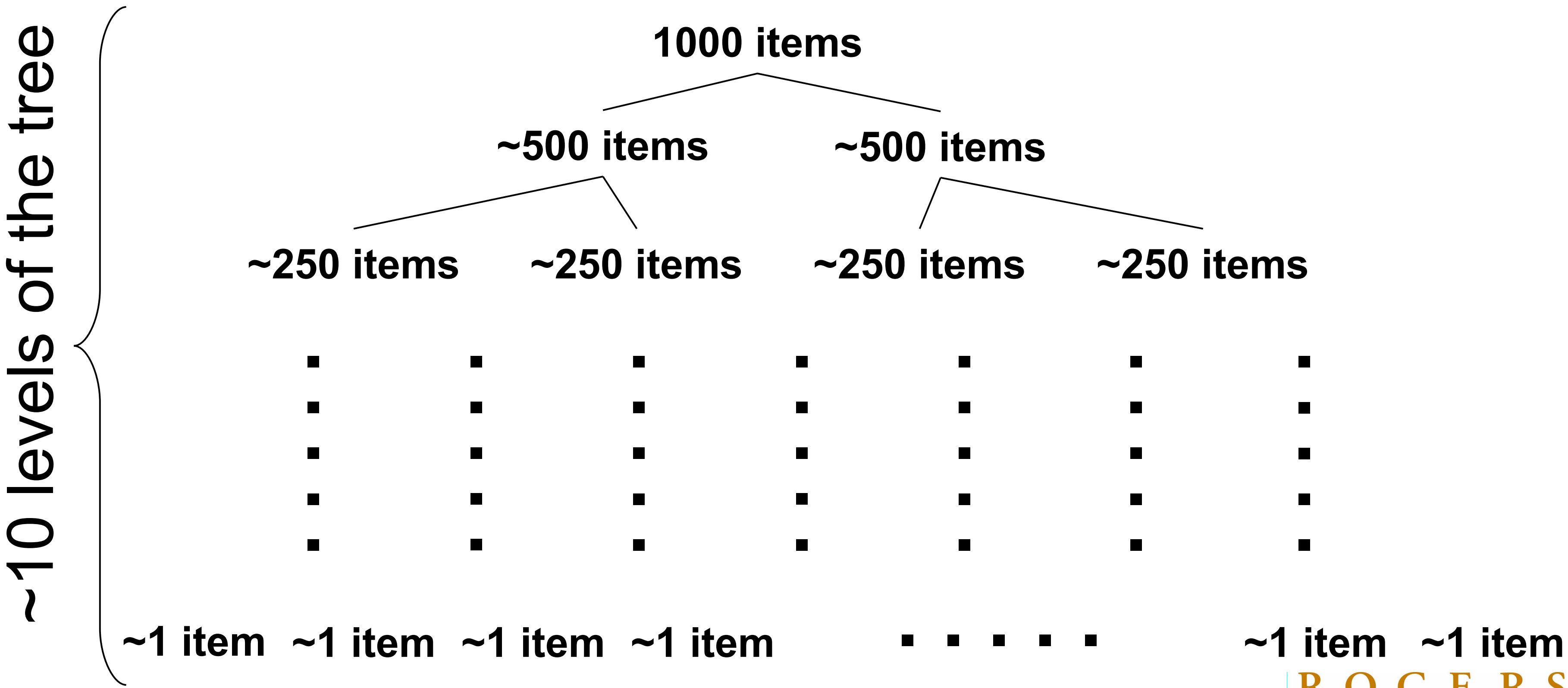
Quicksort

- How many comparisons/swaps?
  - Sort 1000 items:



Quicksort

- How many comparisons/swaps?
  - Sort 1000 items:



Comparing Algorithms

- To sort 1000 times:
  - Quicksort requires ~10,000 comparisons (on average).
  - Bubble sort may require ~500,000 comparisons.
  - Insertion sort may require ~500,000 comparisons.

Sorting Summary

- **Sorting:** key part of many computer programs!!
- **Algorithm:** set of steps for completing a task.
- Different algorithms for same task may have different efficiencies!
- Talked about three sorting algorithms:
  - **Bubble sort, Insertion Sort** and **Quicksort**
- Bubble sort and insertion sort are simple to build, but need **many steps** to sort the list.
- Quicksort is more complex to build, sorts list **much faster**.

ROGERS DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING UNIVERSITY of TORONTO