

- Overview of Today's lecture
 - Infix \Rightarrow Postfix
 - Backtracking
 - Definition of Queue
 - Comparison with Stacks
 - Queue ADT
 - Uses of Queues
 - Queue Applications: Palindromes
 - Queue Applications: Carwash Simulation
- Notes on BasicCalc.java - evaluates fully parenthesized expressions
 - fig 6.5
 - one class BasicCalc
 - * Methods:

```
public static double readAndEvaluate(EasyReader input)
public static void evaluateStackTops(DoubleStack nums, CharStack operations);
```
 - `input.peek()` - look at next character without actually reading (i.e., moving the input pointer)
- Stack Application: Translating Infix to Postfix
 - What is Infix?
 - What is Postfix?
 -
- Program 3 - Translating Infix to Postfix and evaluating.
 - page 322 number 8
 - fig 6.9, 6.10
 -
- Backtracking
 - moves with a lot of choices
 - push moves onto stack
- n-queens problem

He who comes first, eats first.
- Definition
 - A queue is a First-In/ First-Out (FIFO) data structure
 - e.g., waiting in line for a bank teller

- We add the rear and serve(remove) from the front

- Comparison with Stacks

| | |
|-------|-------|
| STACK | QUEUE |
| LIFO | FIFO |

- Queue ADT

- insert
- getFront - get and remove the first element
underflow *Rightarrow* NoSuchElementException
EmptyStackException for stacks, but none for queues
- isEmpty()
- peek()

- Uses for queues

- simulations:
- Input/Output buffering

INPUT

OUTPUT

- In general saving data for later processing

- Self-Test Page 329 question 3

Write pseudocode that will read an even number of characters and write the odd characters(1st, 3rd, ...) on one line and then write the even characters (2nd, 4th, ...) on a second line.

- Queue Applications: Palindromes

- What is a palindrome?
- Example:
- What happens if you push a sequence of char onto a stack then take them off? (order)?

- palindrome test -
 1. read char until the end of a line
 2. As read push copies of chars onto a stack and a queue
 3. then pop and getFront and compare until stack/queue are empty
- Notes on Palindrome.java
 - fig 7.3
 - io.EasyReader //Appendix B again
 - CharQueue from edu.colorado.collections.CharQueue
 - !stdin.isEOLN()
- Queue Applications: Discrete Event Simulation
 - a time step is fixed
 - while (time \leq TimeToSimulate)
 - the clock is advanced
 - use random numbers to determine what happens (generate events) during that time slot
- Car Wash Simulation
 - automated carwash
 - settings range 1 to 10 minutes:
 - Fast (1 minute) more expensive uses more soap
 - Run simulation allowing variation of settings to see:
 - * how much profit
 - * how many cars back up
 - Note random arrival uses Math.random()
- Car Wash Specification

INPUT

- amount of time needed to wash 1 to 10
- probability of a car arriving during next second
- total time to simulate

OUTPUT

- number of cars
- average time in line till washed
- ???

- Car Wash Design

OBJECTS

- cars in queues - What data is needed here?

- Washer -
- BooleanSource - provides a random sequence of booleans for car arrivals
- Statistician (averager)
- pseudocode fig 7.5
- Lab 7 - n queens problem
 - Can you place n queens on an n by n chessboard so that none attack each other? i.e., none are on the same row, column, or diagonal?
 - Powerpoint slides
 - Backtracking with stack of positions
 -