What is CS1100 all about?

What is CS1100 all about?

Problems and how to solve them

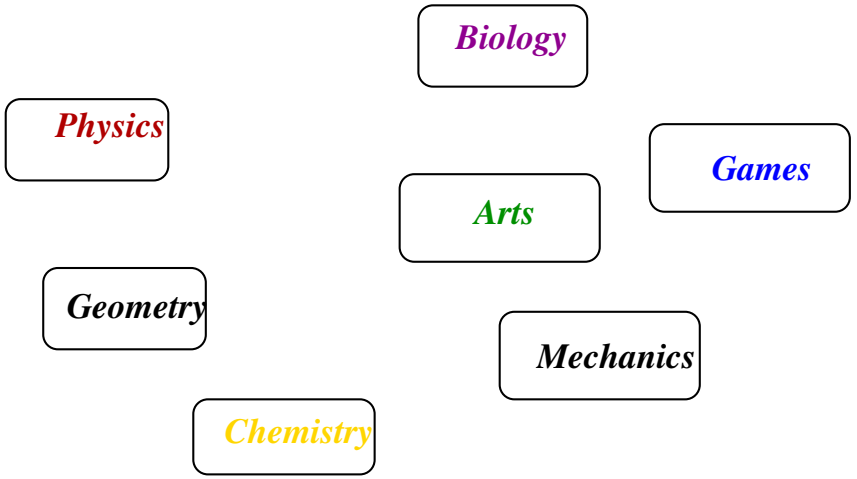
What is CS1100 all about?

Problems and how to solve them using a computer.

What is CS1100 all about?

Problems and how to solve them using a computer.

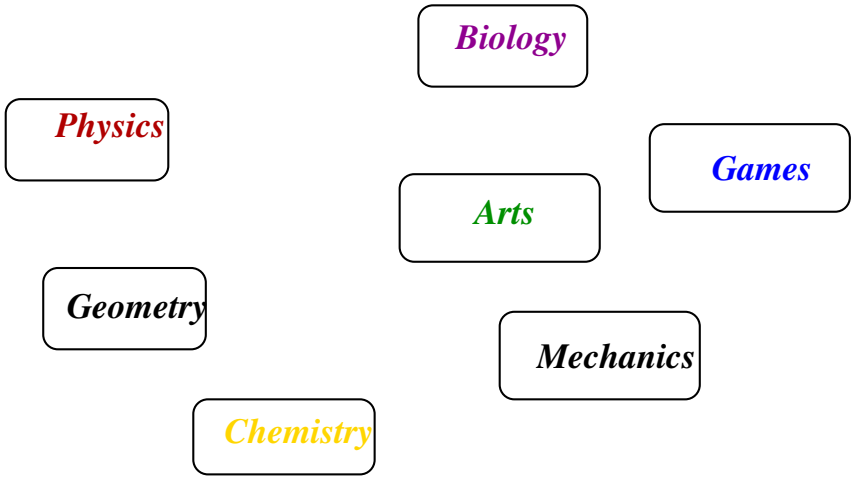
What is the source of the problems?



What is CS1100 all about?

Problems and how to solve them using a computer.

What is the source of the problems?

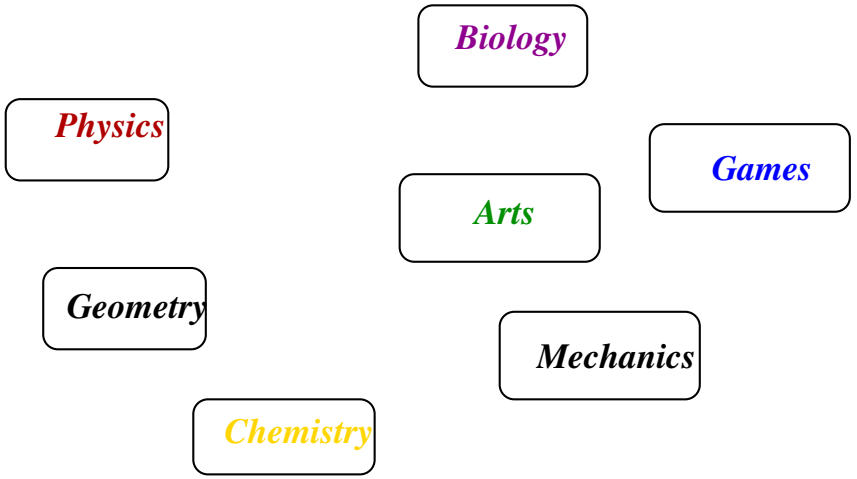


Our goal: Understand the problem, formulate a solution and ask the computer to do it!

What is CS1100 all about?

Problems and how to solve them using a computer.

What is the source of the problems?



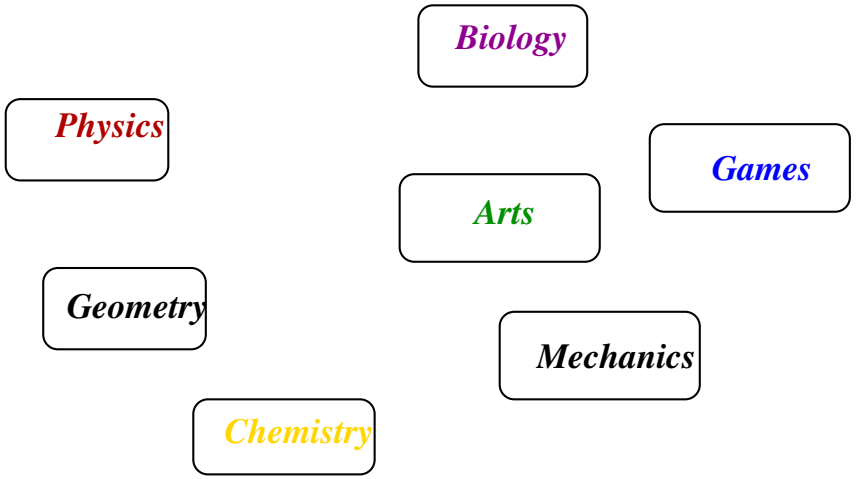
Our goal: Understand the problem, formulate a solution and ask the computer to do it!

If you want to learn something, teach it

What is CS1100 all about?

Problems and how to solve them using a computer.

What is the source of the problems?



Our goal: Understand the problem, formulate a solution and ask the computer to do it!

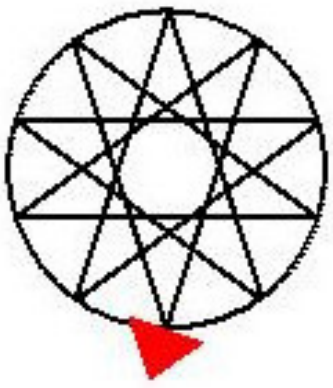
If you want to learn something, teach it to a computer

Example #1 : Drawing Patterns

Learnings:

- A simple but useful language – turtle graphics.
- Using "repeat" to achieve complicated but repetitive tasks.

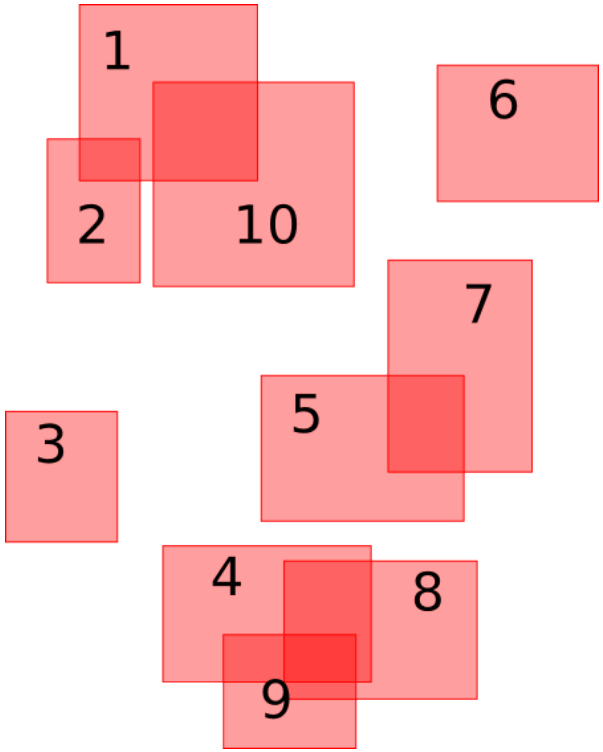
Example #1 : Drawing Patterns



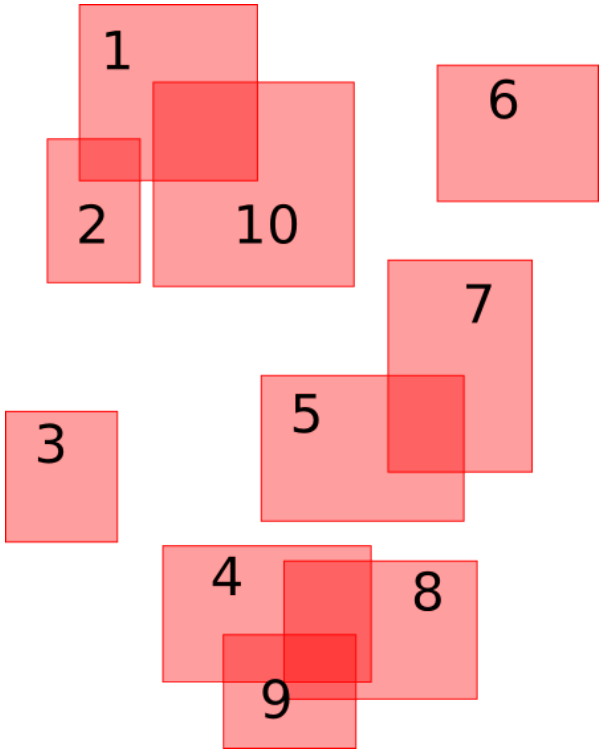
Learnings:

- A simple but useful language – turtle graphics.
- Using "repeat" to achieve complicated but repetitive tasks.
- The power of control modification – via loops.

Example #2 : Geometric Problems



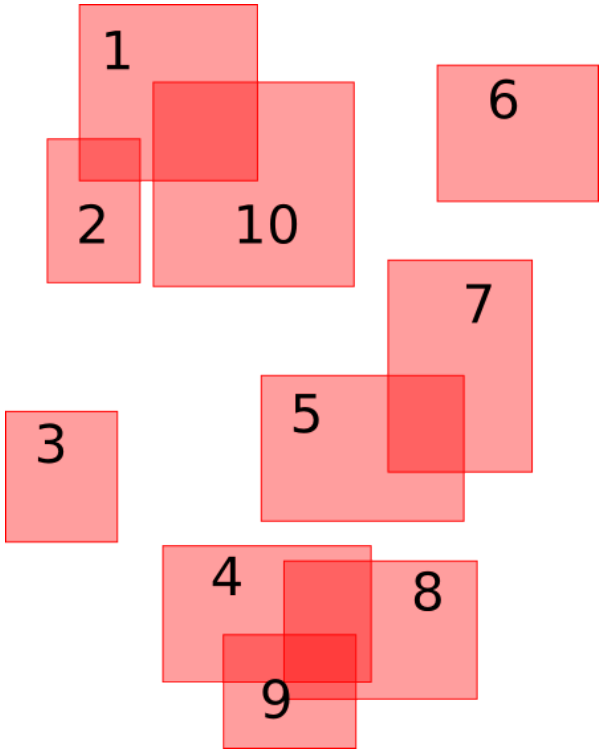
Example #2 : Geometric Problems



Some questions:

- Are two rectangles intersecting?
- Drawing triangles of given dimensions.
- Computing areas using alternate formulas.

Example #2 : Geometric Problems



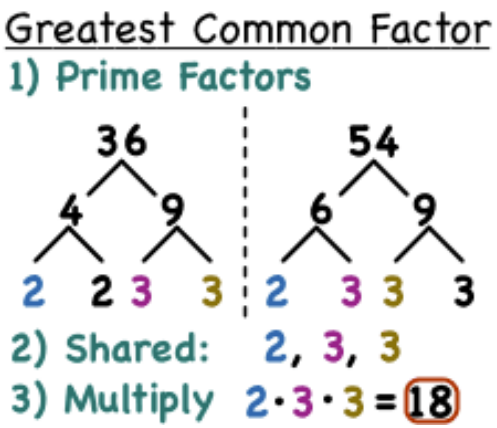
Some questions:

- Are two rectangles intersecting?
- Drawing triangles of given dimensions.
- Computing areas using alternate formulas.

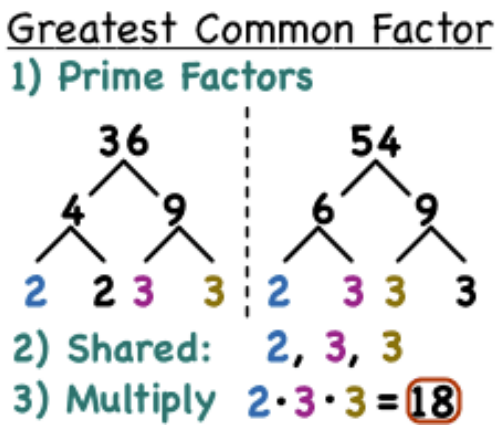
Learnings:

- Logical approach to the problems.
- Formalizing known concepts as precise statements in a prog. language.

Example #3 : Number theoretic questions



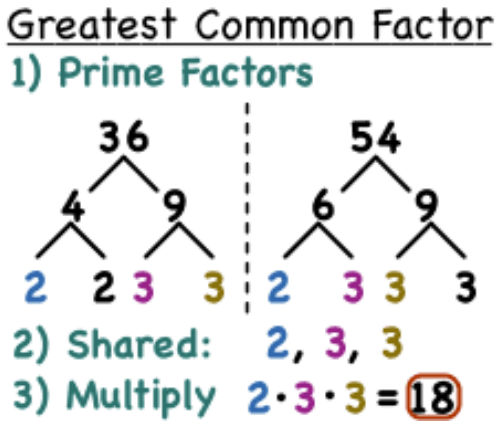
Example #3 : Number theoretic questions



Some questions:

- Is a number prime? Output prime factors.
- Compute GCD / LCM of two numbers.

Example #3 : Number theoretic questions



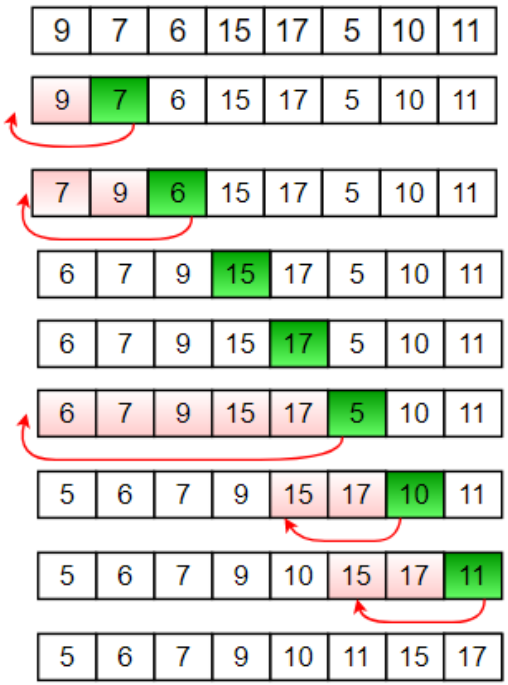
Some questions:

- Is a number prime? Output prime factors.
- Compute GCD / LCM of two numbers.

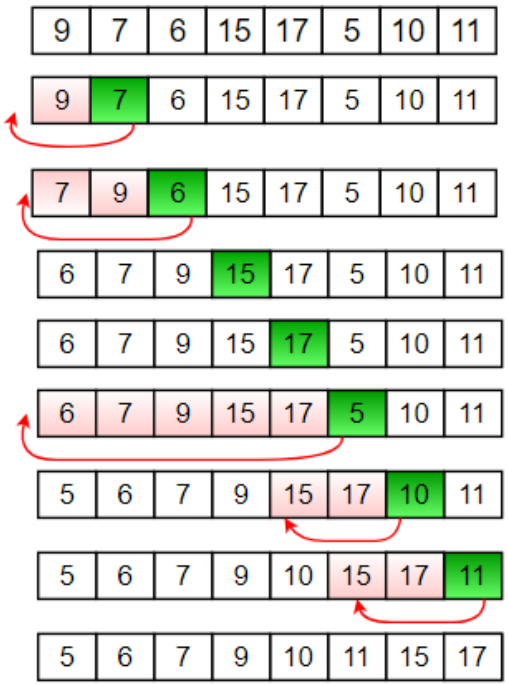
Learnings:

- Again power of being able to repeat.
- Reducing the problem to smaller size.
- A efficient algorithm can make a significant difference!

Example #4 : Searching and Sorting



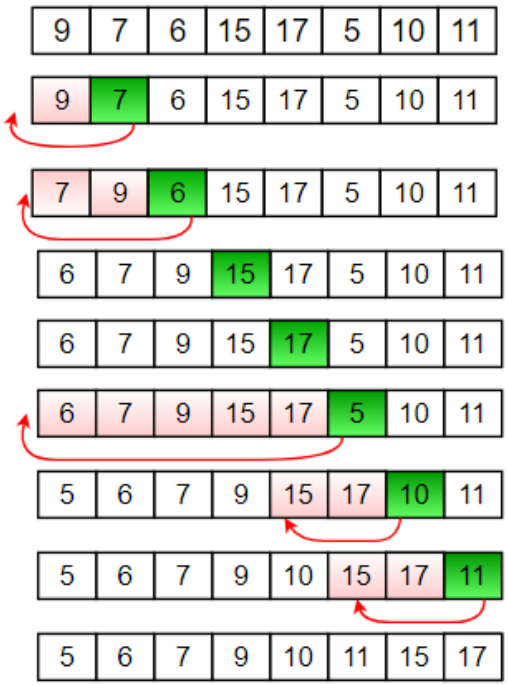
Example #4 : Searching and Sorting



Some questions:

- How to search when list is sorted?
- Sort a list of elements.

Example #4 : Searching and Sorting



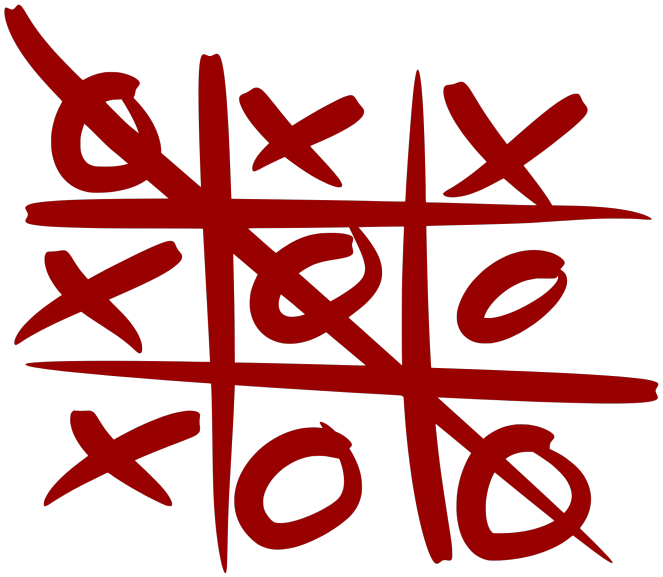
Some questions:

- How to search when list is sorted?
- Sort a list of elements.

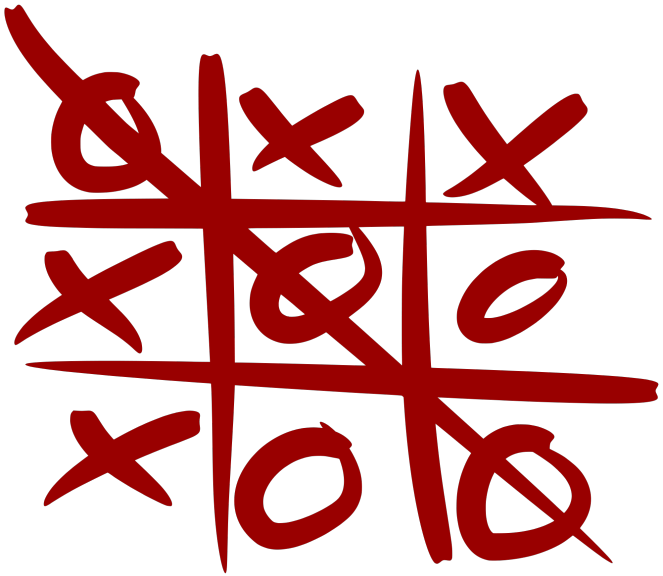
Learnings:

- Binary versus Linear search.
- Different ways to sort – again converting known ideas to precise statements.
- A efficient algorithm can make a significant difference!

Example #5 : Tic-Tac-Toe



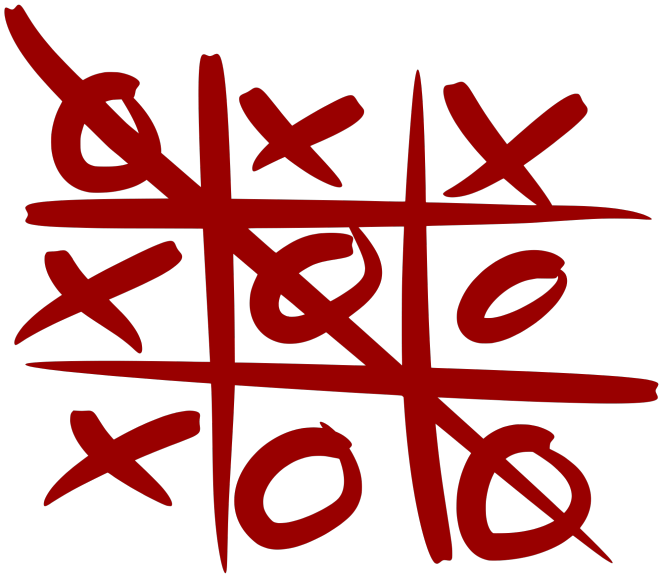
Example #5 : Tic-Tac-Toe



Some questions:

- Tic-Tac-Toe.
- Word Grids – finding strings.

Example #5 : Tic-Tac-Toe



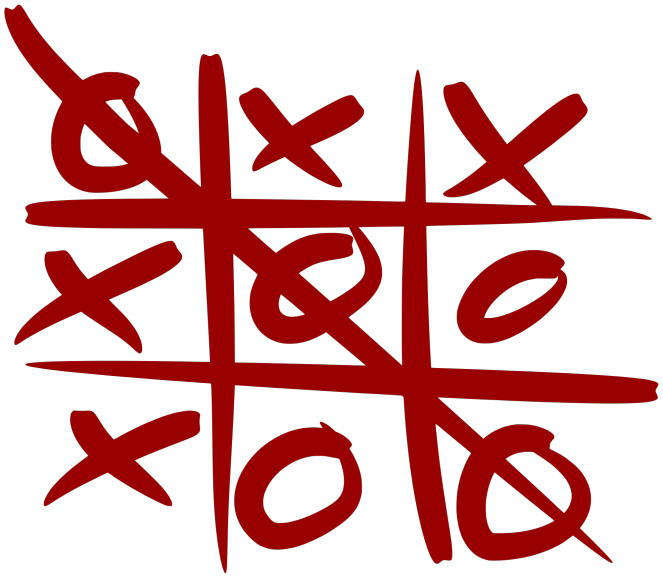
Some questions:

- Tic-Tac-Toe.
- Word Grids – finding strings.

Learnings:

- Breaking problems into subtasks.
- Modular way of thinking.

Example #5 : Tic-Tac-Toe



Some questions:

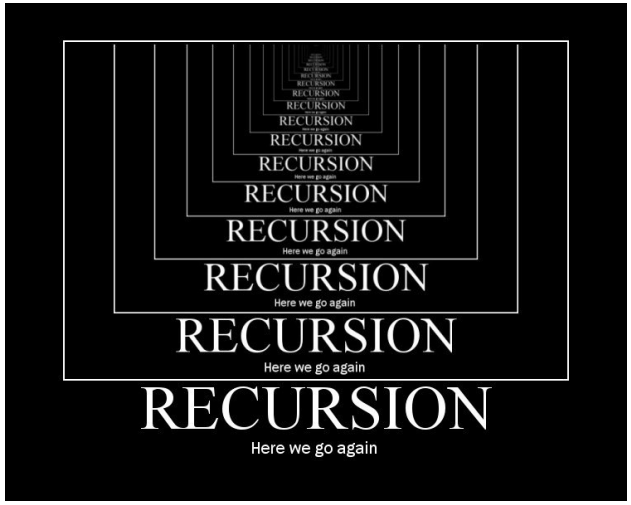
- Tic-Tac-Toe.
- Word Grids – finding strings.

Learnings:

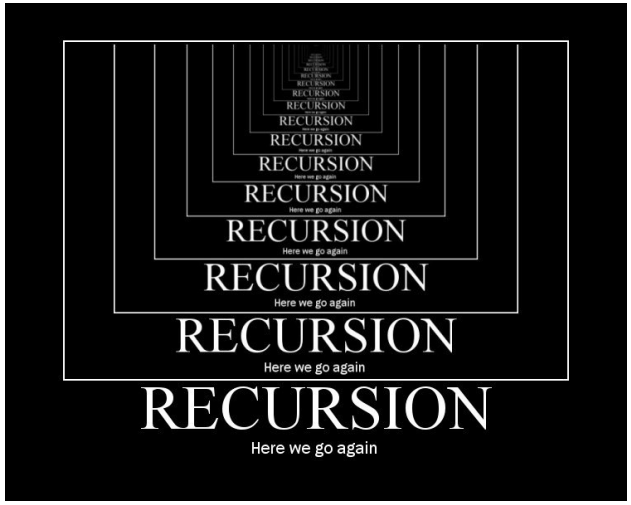
- Breaking problems into subtasks.
- Modular way of thinking.

Did not consider strategy games!

Concept #6 : Recursion



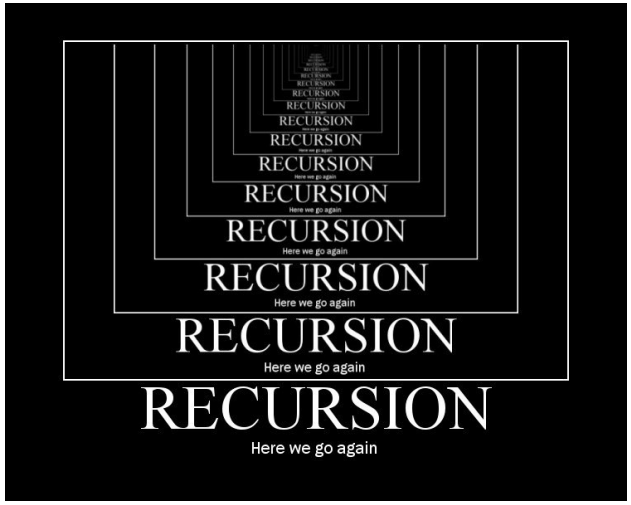
Concept #6 : Recursion



Some questions:

- Strings – palindromes, reversals.
- Useful data structures – queues, stacks.

Concept #6 : Recursion



Some questions:

- Strings – palindromes, reversals.
- Useful data structures – queues, stacks.

Learnings:

- Thinking recursively.
- Breaking down tasks into sub-tasks of the same type!

CS1100 – Problem Solving Using Computers

- Data Types in C, Operators. Input and the Output.
- Modifying the control flow in programs `if-else`, `switch`, `while`, `do-while`, `for`.
- `Arrays` and `Strings` in C.
- Functions & Modular programming, Recursion.
- Pointers, Pass by reference, Pointer arithmetic.
- Structures in C.
- Structures and Pointers.

Week 1-15

CS1100 – Problem Solving Using Computers

- Data Types in C, Operators. Input and the Output.
- Modifying the control flow in programs `if-else`, `switch`, `while`, `do-while`, `for`.
- `Arrays` and `Strings` in C.
- Functions & Modular programming, Recursion.
- Pointers, Pass by reference, Pointer arithmetic.
- Structures in C.
- Structures and Pointers.

Week 1-15

The 3 Ds: Design, Debug, Detect

Final Take Away