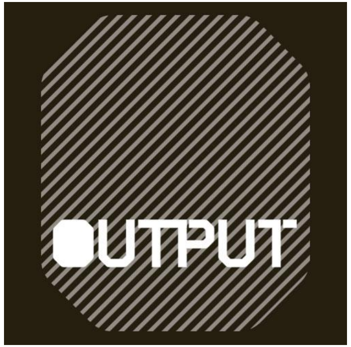
Pseudocode

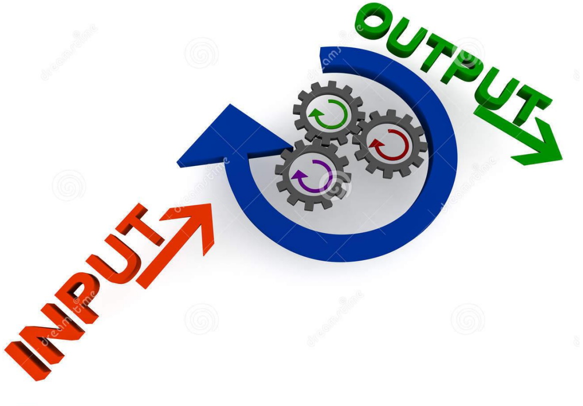
Pseudocode

- Describes algorithms in a more robust, mathematical sense then writing them step-by-step

Pseudocode

- Needs defined: Inputs, Outputs, and Assumptions
    - Help clarify what is needed and what the goal is





Decimal to Hex: Pseudocode

- Converting base 10 to hex- **Divide by 16 Algorithm**

1) Assume the number is > 0
2) Divide the number by 16, write the remainder in a stack (bottom up)- convert to hex notation if necessary
3) When the number is reduced to zero, flip the stack. This is your hex number.

Decimal to Hex: Pseudocode

- Assumptions: Number is > 0
- Input: A base 10 number
- Output: A hexadecimal number

- *These three parts MUST be included in any pseudocode you write*

Pseudocode Details

- Prompts must appear as they would appear to a user
- State destination of output (ex: Display, File)
- Each number should be a variable
  - Should represent a GENERAL number, not a specific value

Pseudocode Details

- Surround variable names with ‘<’ and ‘>’
- Make up instructions as necessary
  - “Turn on the computer”, “Vibrate phone for 1 second”

Decimal to Hex Pseudocode

```
<dec> = input

<stack> = <empty>

While <dec> does not = 0:

    <dec> = <dec> / 16

    Push remainder to top of stack

Reverse <stack>

Print <stack> to Display
```