STEAM CLOWN™ PRODUCTIONS

FLOW CHART & ALGORITHMS

Planning Your Code

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

STEAM CLOWN™ PRODUCTIONS

SEE APPENDIX A, FOR LICENSING & ATTRIBUTION INFORMATION

by-nc-sa-3.0

https://creativecommons.org/licenses/by-nc-sa/3.0/

https://creativecommons.org/faq/#what-does-some-rights-reserved-mean

STEAM CLOWN™ PRODUCTIONS

CAN I GET A COPY OF THESE SLIDES? YES, PROBABLY…

Look Here – <Add Link>

Or, most presentation lecture slides can be found indexed on www.steamclown.org and maybe blogged about here on Jim The STEAM Clown's Blog, where you can search for the presentation title. While you are there, sign up for email updates

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

Flowchart Introduction

- The first flowchart was created by **Von Neumann** in 1945
- Flowchart are important for planning and working of a program
- Flowcharts decreases our efforts i.e. they are easy to understand and check logics and algorithms
- Flowcharts are important to do before you start writing code

STEAM Clown™ & Squeaky Hinge Productions

What you will know…

- Prior Knowledge
  - How…
  - How…

- What You Will Know & Be Able To Do
  - Calculate…
  - Build a…
  - Be able to describe… how an R/C circuit works

STEAM Clown™ & Squeaky Hinge Productions

How will you be measured

- To what extent will the verb be measured?
- How will success be determined?

NEW WORDS...

- Algorithm
- Flowchart
- Graphical
- Pseudocode

ALGORITHMS

- It is a list of instructions specifying a precise description of a step by step process that terminates after a finite number of steps for solving an algorithm problem producing the correct answer in the end
- It is a recipe for solving problems
- A finite set of an instruction that specifies a sequence of operation to be carried out in order to solve a specific problem.
- An unambiguous procedure specifying a finite number of steps to be taken.

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

What’s another description for an Algorithms?

Flowcharts

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

What is a Flowchart?

- Flowcharts are a graphical representation of a program
  - Representation of a solution to a particular problem, which comes under the category of Programming Practices and Techniques
- Flowcharts are symbolic diagrams which shows type of data(numeric, character etc.), data flow, control flow and programming logics and algorithms.

Programming an Algorithm

- A typical programming task can be divided into
- two phases:
  - ***Problem solving phase***
    - Produce an ordered sequence of steps that describe solution of problem
    - This sequence of steps is called an ***algorithm***
  - ***Implementation phase***
    - implement the program in some programming language

STEAM Clown™ & Squeaky Hinge Productions

ALGORITHM – PROBLEM SOLVING PHASE

- PSEUDOCODE - specifies the steps of algorithm using natural language of superimposed control structure.
- FLOWCHART - a traditional graphical tool with standardized symbols. Show the sequence of steps in an algorithm

PROPERTIES OF ALGORITHM

- **Finiteness** – Has an exact number of steps to be taken and has an end
- **Absence of Ambiguity** - means that every instruction is precisely described and clearly specified
- **Sequence of Execution** - instructions are performed from top to bottom
- **Input and Output** - defined the unknowns of the problem is specified and with the expected outcome
- **Effectiveness** - the solution prescribed is guaranteed to give a correct answer and that the specified process is faithfully carried out
- **Scope Definition** - applies to a specific problem or class of problem

Before You Start Writing Code...

- **State the problem clearly**
  - A problem cannot be solved correctly unless it is understood
- **Plan and Write the Logical Order of Instructions**
  - Refine the algorithm successively to get step by step detailed algorithm
  - Get description that is very close to a computer language
  - The computer follows the direction exactly at the given sequence.

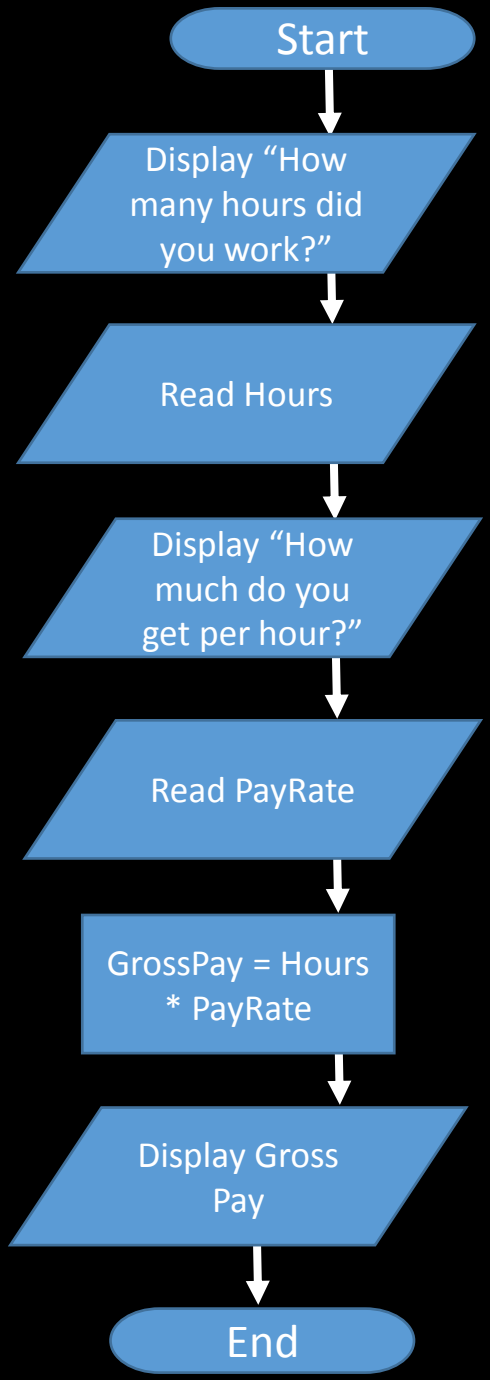
STEAM Clown™ & Squeaky Hinge Productions

FLOWCHARTING GUIDLINES

- The flowchart should flow from top to bottom
- If the chart becomes complex, utilize connecting blocks
- Avoid intersecting flow lines
- Use meaningful description in the symbol

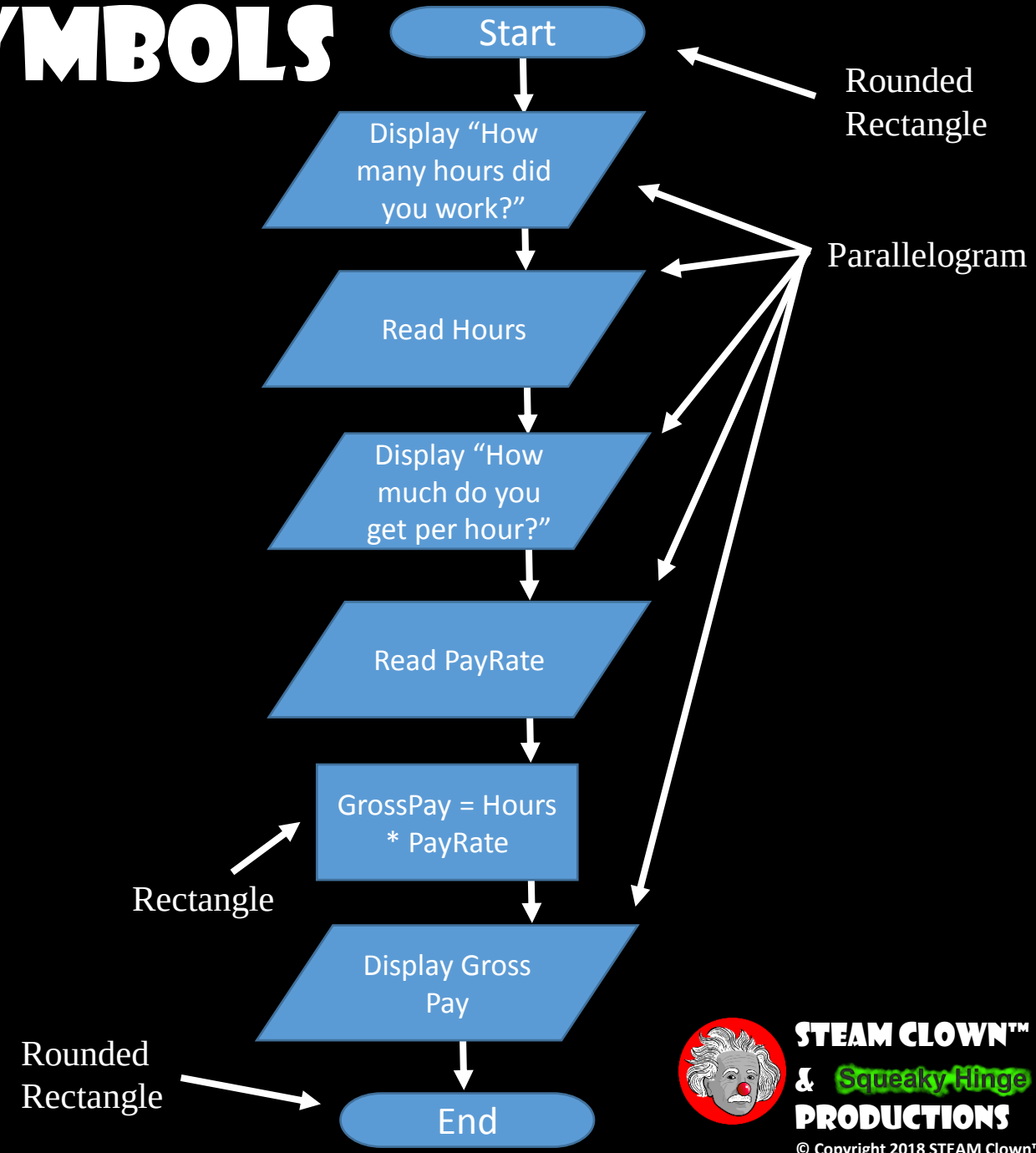
Flowchart Example

- A flowchart is a diagram that depicts the “flow of control” of a program.



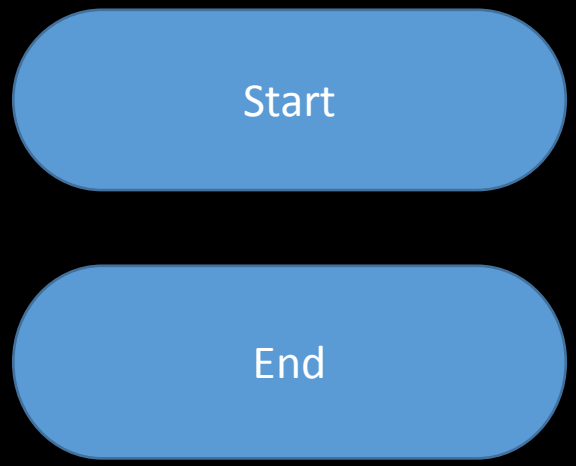
Basic Flowchart Symbols

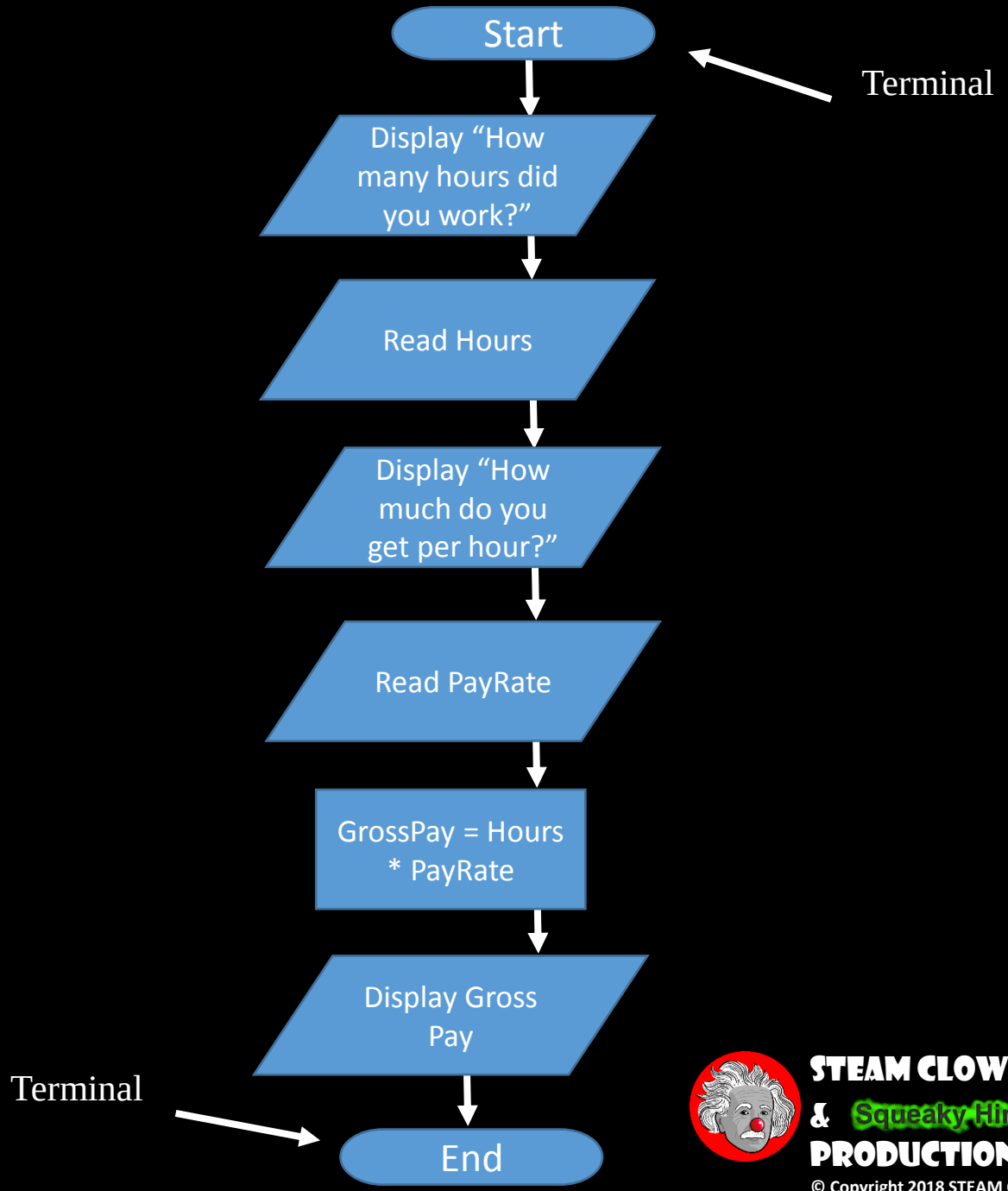
- Notice there are three types of symbols in this flowchart:
  - rounded rectangles
  - parallelograms
  - a rectangle
- Each symbol represents a different type of operation.



TERMINALS

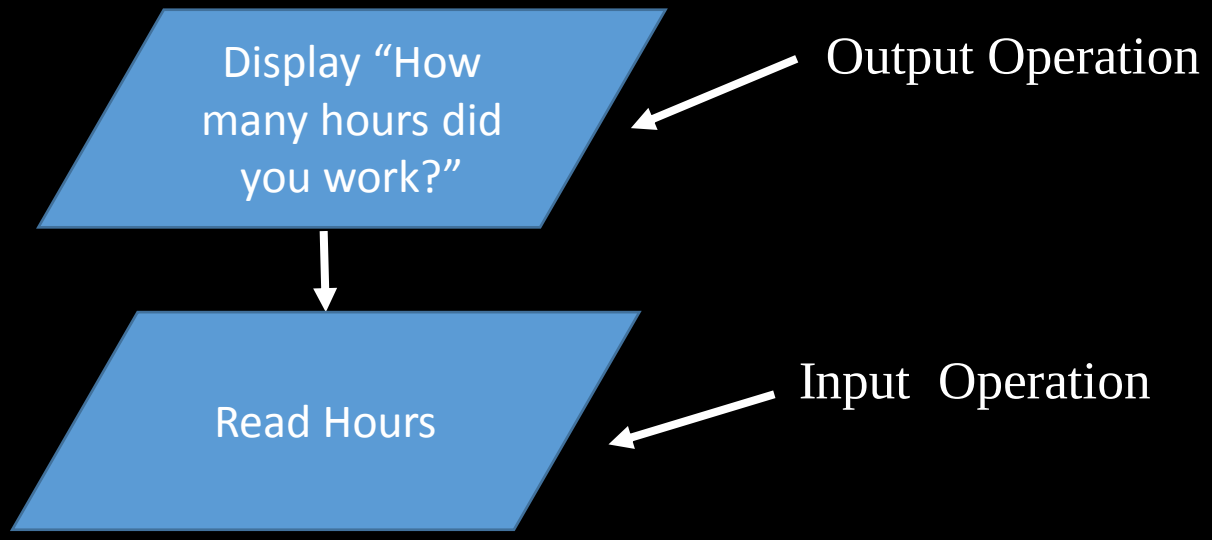
- Represented by rounded rectangles
- Indicate a starting or ending point

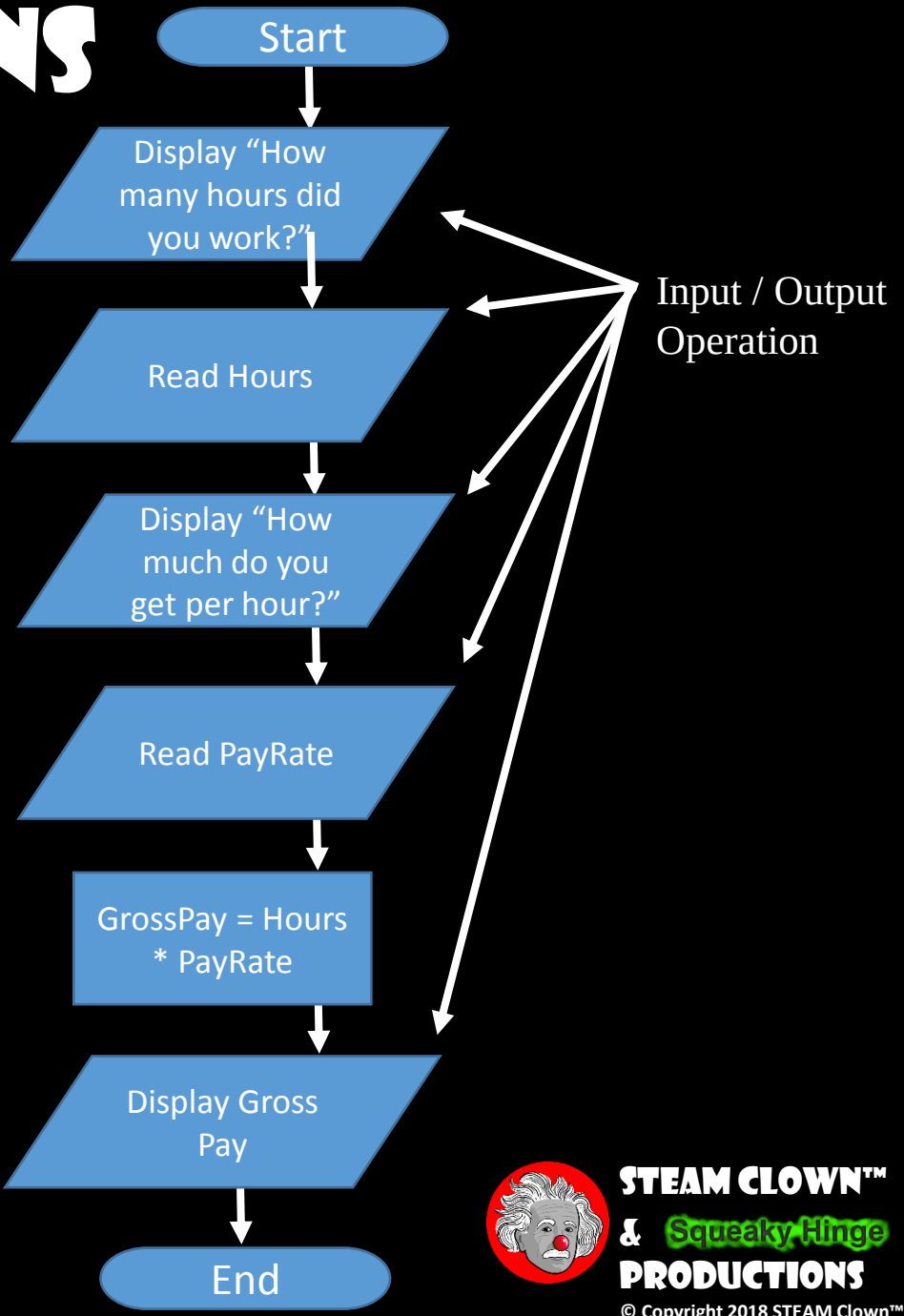




INPUT/OUTPUT OPERATIONS

- represented by parallelograms
- indicate an input or output operation

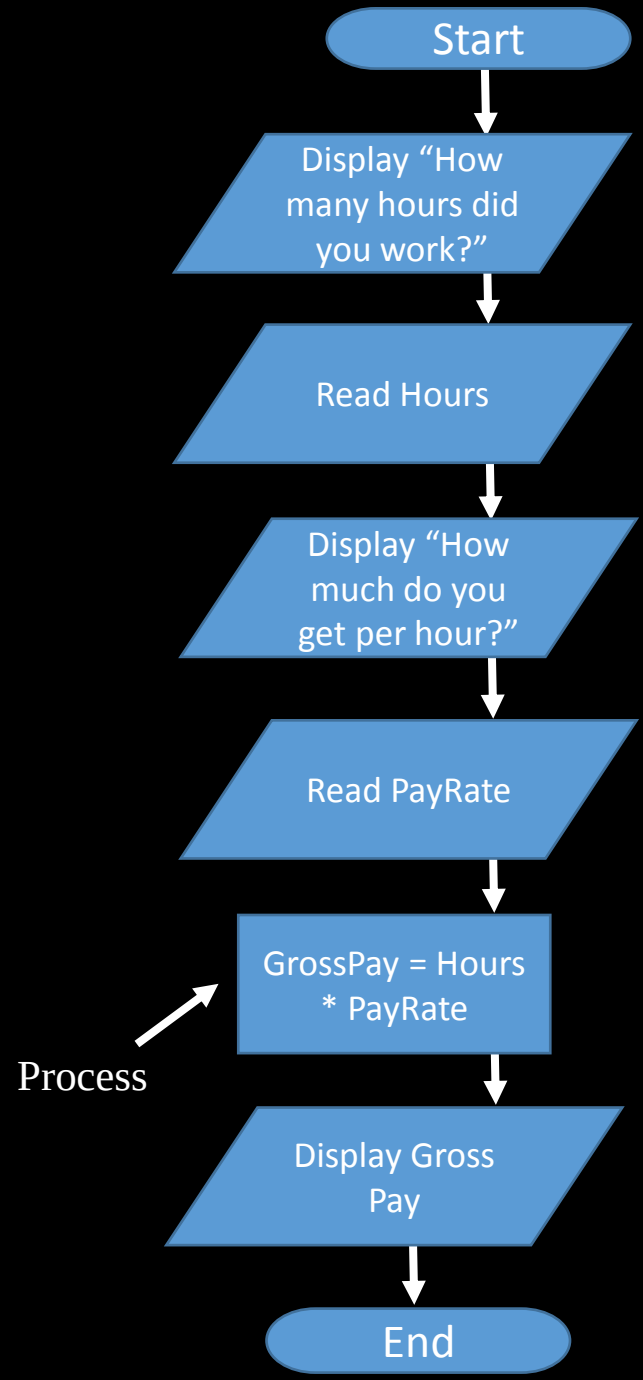


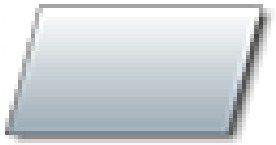


PROCESS

- represented by rectangles
- indicates a process such as a mathematical computation or variable assignment

GrossPay = Hours * PayRate



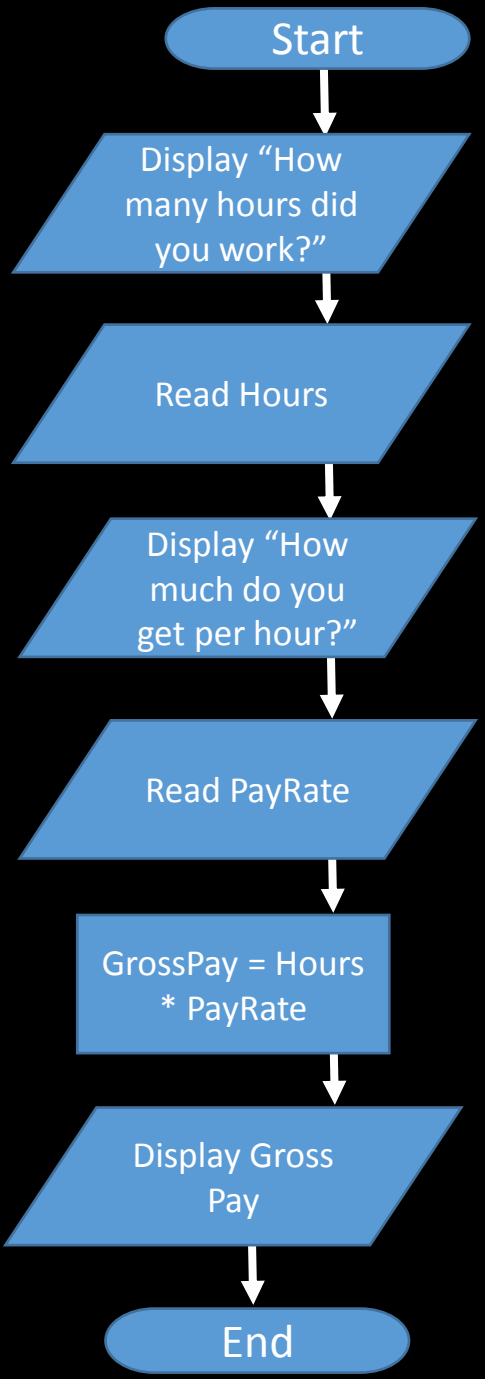
Symbol	Name	Function
	Start/end	An oval represents a start or end point.
	Arrows	A line is a connector that shows relationships between the representative shapes.
	Input/Output	A parallelogram represents input or ouptut.
	Process	A rectangle represents a process.
	Decision	A diamond indicates a decision.

We will look at "Decisions" in a few slides

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

Stepping Through the Flowchart

- Identify the Inputs and Outputs
- List Variables



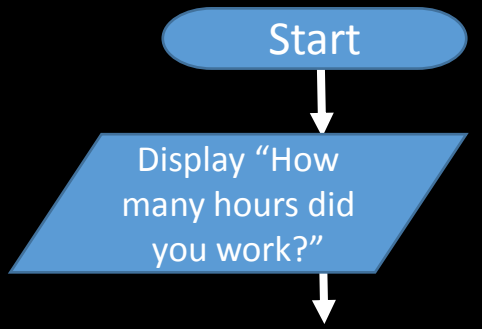
Variables & Their Values

Hours = ??

PayRate = ??

GrossPay = ??

Stepping Through the Flowchart



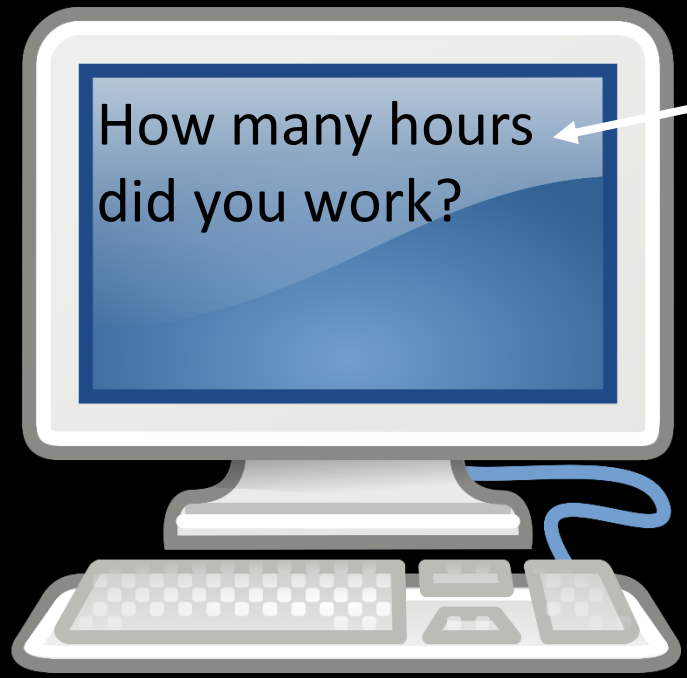
- Output the question "Hours" to the users

Variables & Their Values

Hours = ??

PayRate = ??

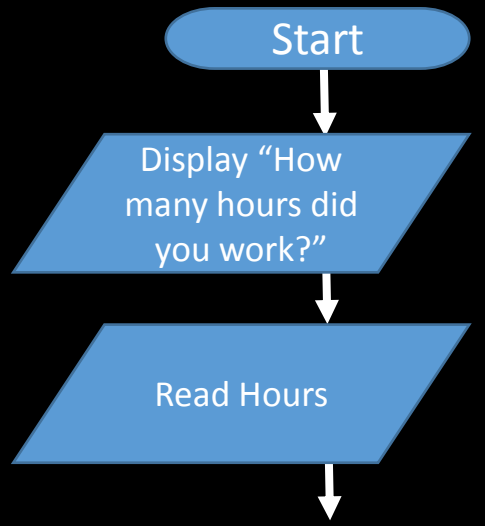
GrossPay = ??



Program Outputs "How many hours did you work?"

Stepping Through the Flowchart

- Get Answer “Hours” from the users

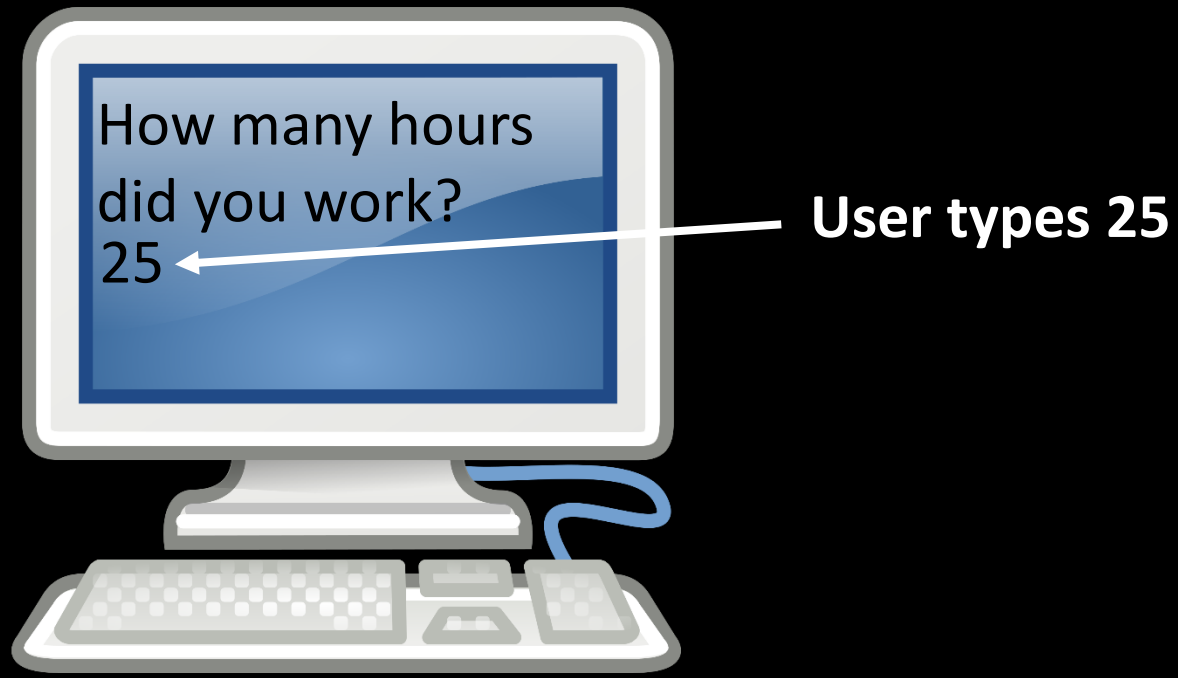


Variables & Their Values

Hours = 25

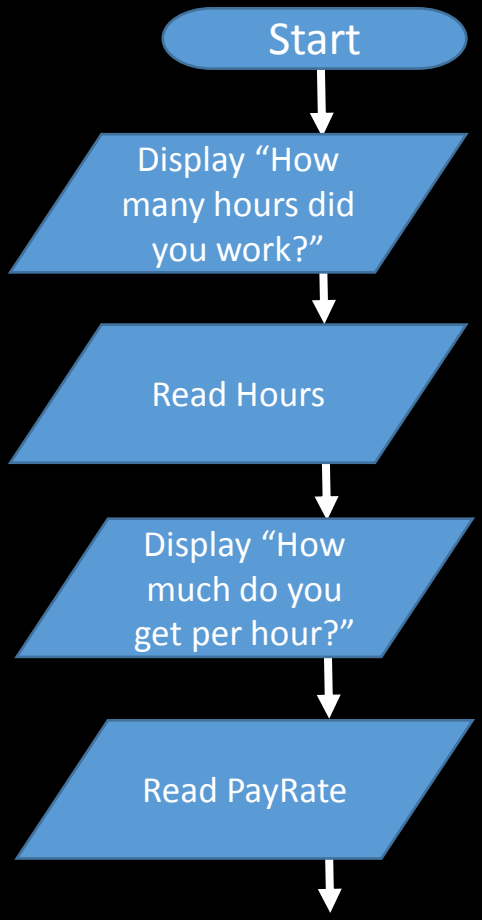
PayRate = ??

GrossPay = ??



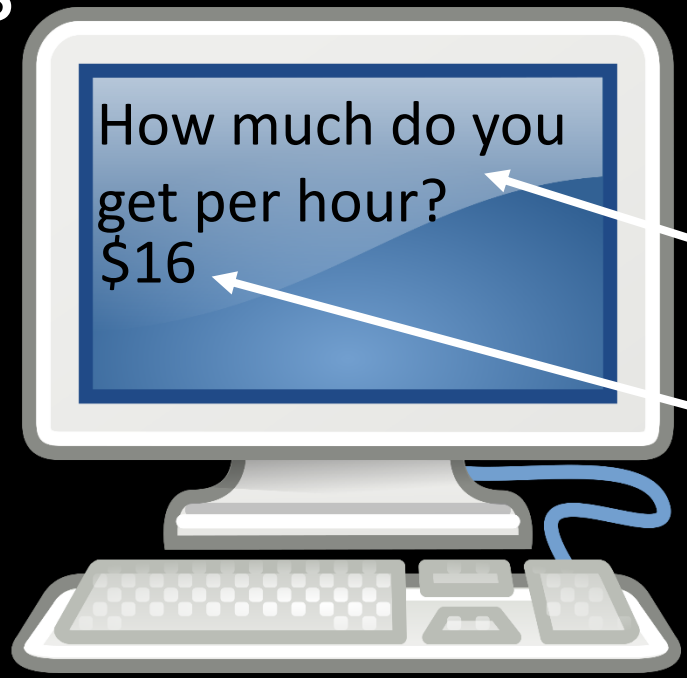
Stepping Through the Flowchart

- Output the question "PayRate" to the users
- Get Answer "PayRate" from the users



Variables & Their Values

Hours = 25
PayRate = $16
GrossPay = ??



Program Outputs "How do you get?"

Get PayRate from User

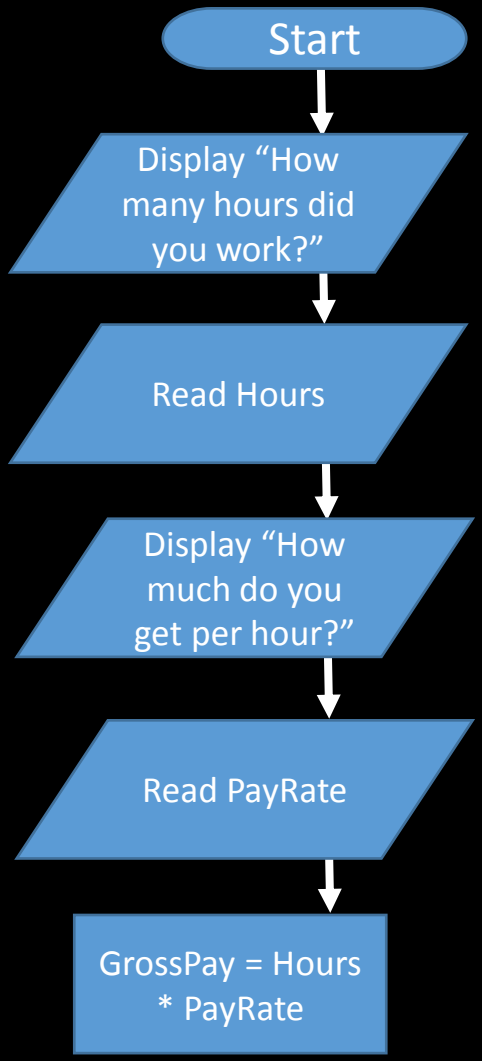
STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

Stepping Through the Flowchart

- Process the calculation

GrossPay = Hours * PayRate

GrossPay = 25 * $16

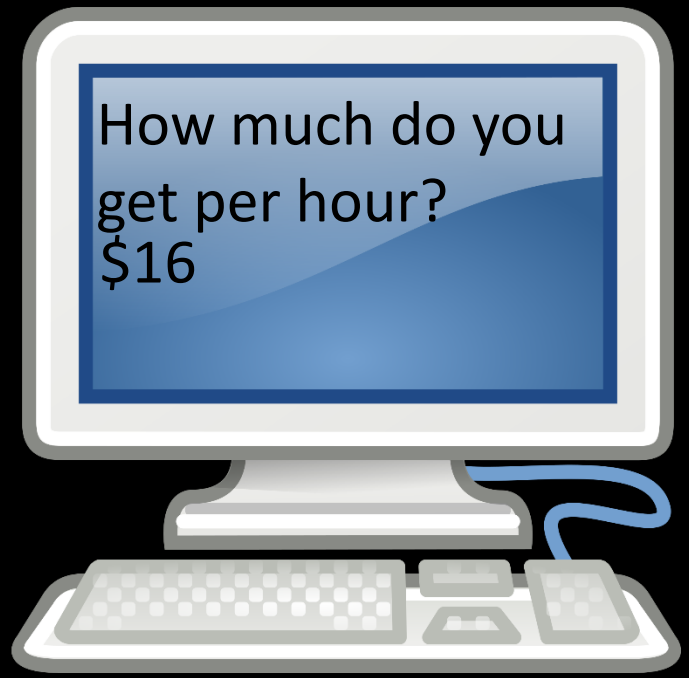


Variables & Their Values

Hours = 25

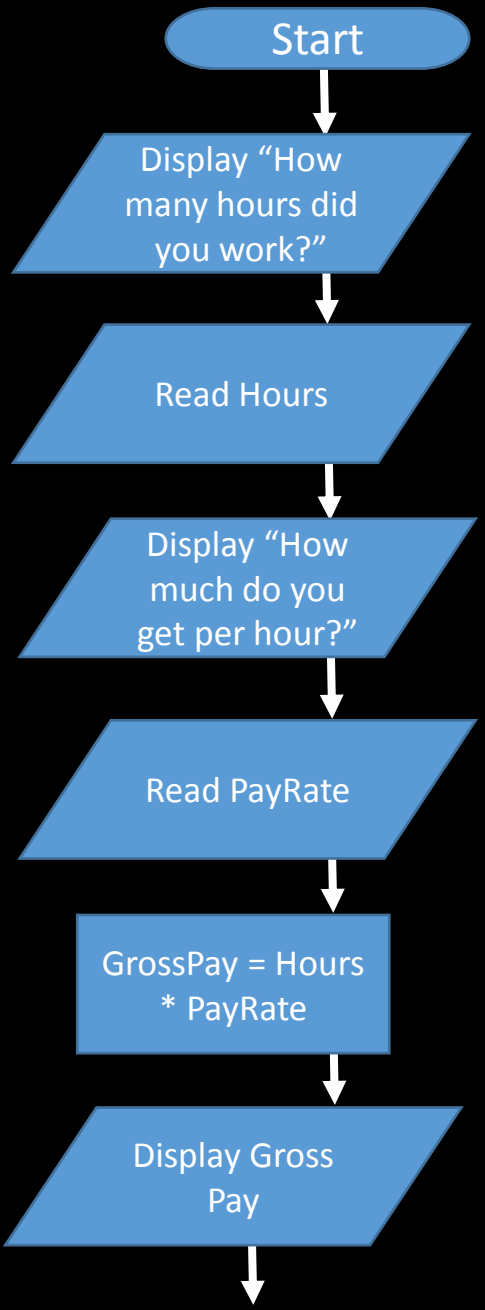
PayRate = $16

GrossPay = $400



Stepping Through the Flowchart

- Output the "GrossPay" to the users



Variables & Their Values

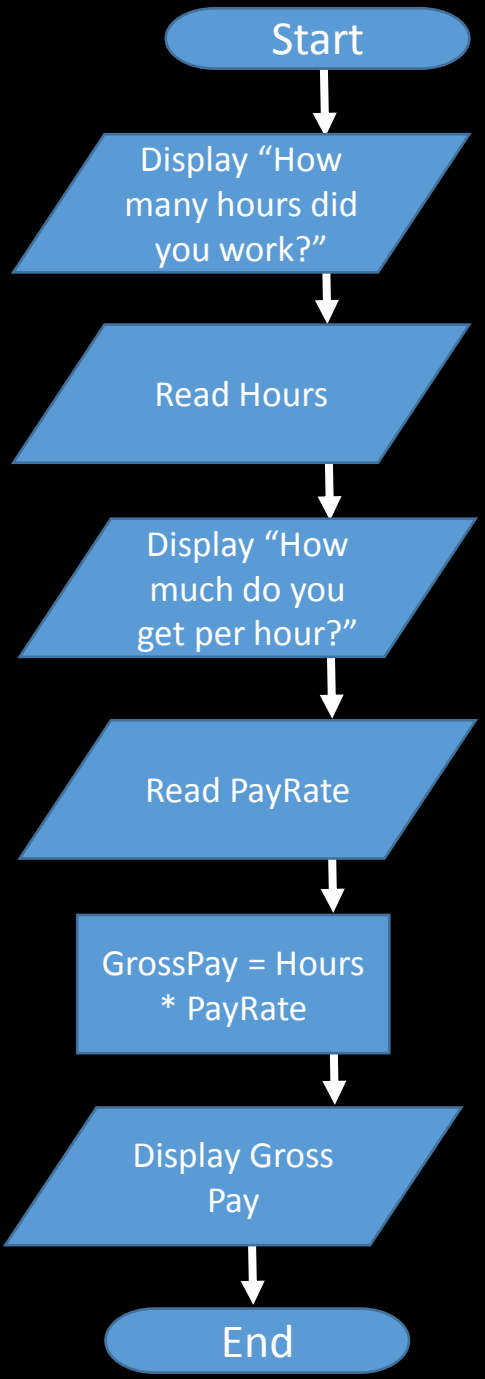
Hours = 25
PayRate = $16
GrossPay = $400

Stepping Through the Flowchart

- End

Variables & Their Values

Hours = 25
PayRate = $16
GrossPay = $400



Three Flowchart Structures

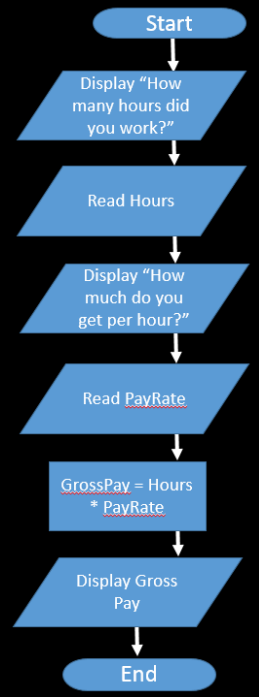
- Sequence
- Selection
- Iteration

What do you think these mean?

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

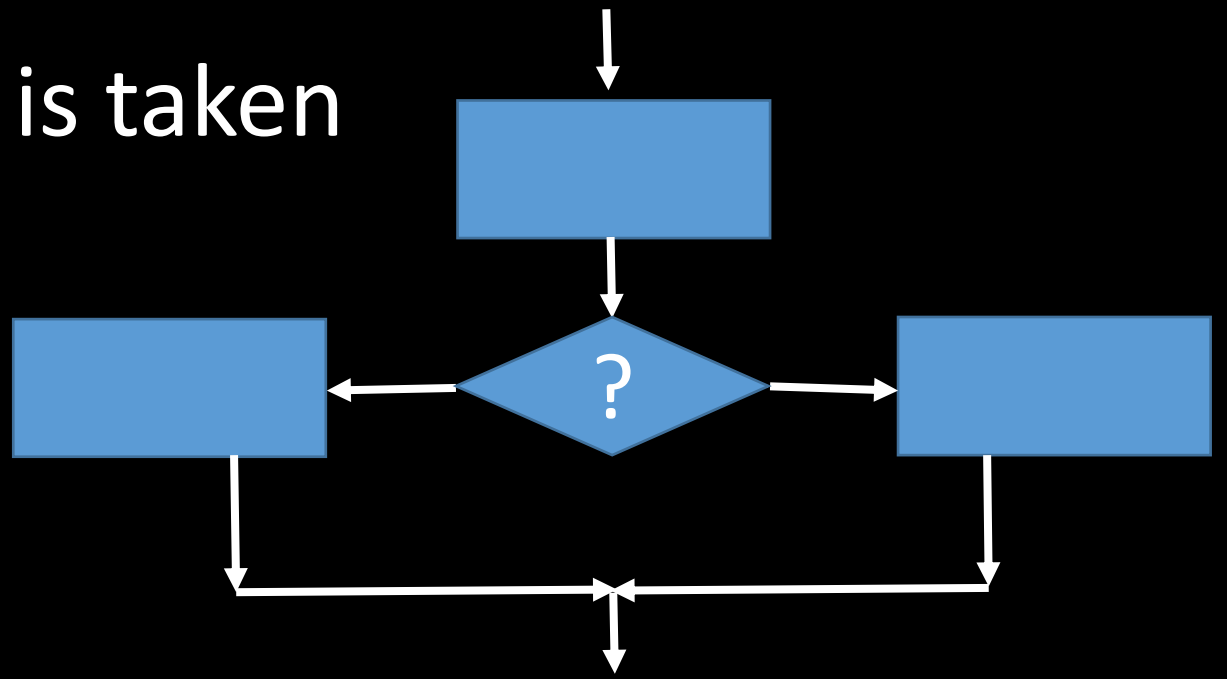
SEQUENCE STRUCTURES

- A series of actions are performed in a sequence
- The pay calculation example was a sequence flowchart
  - There are not “decisions” or selections in a top level sequence



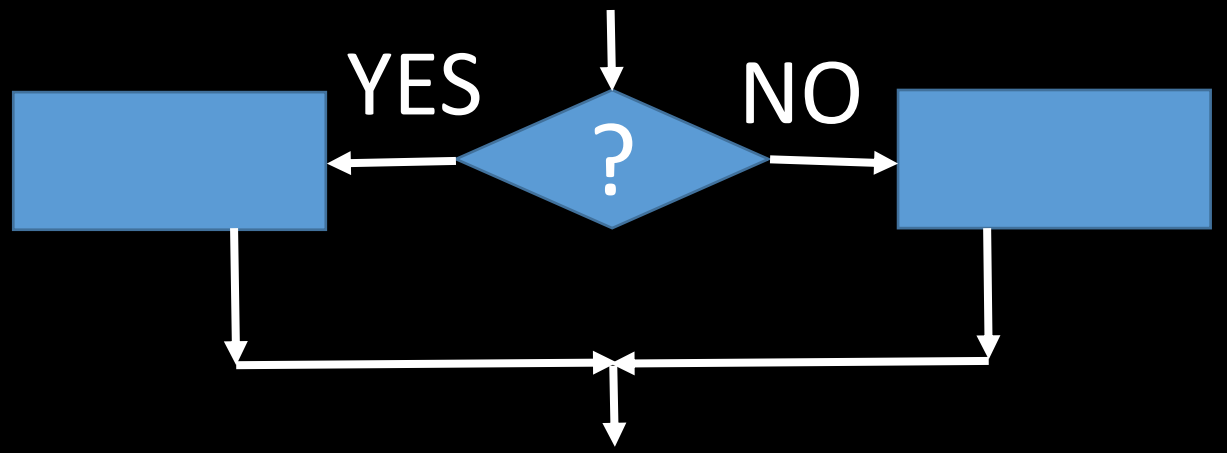
Selection Structure

- This where a question is being asked
- This is a decision point
- One of two possible actions is taken
  - This depends on the result of the test condition
  - Typically is a “TRUE” or “FALSE” answer



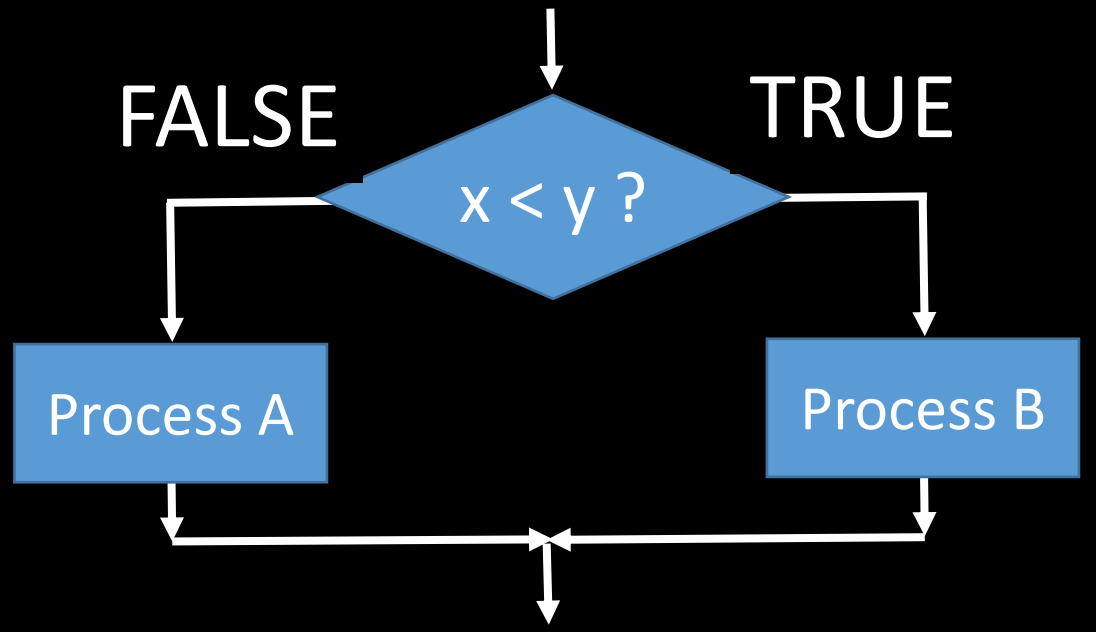
Selection Structure

- A new symbol, the diamond, indicates a yes/no question. If the answer to the question is yes, the flow follows one path. If the answer is no, the flow follows another path



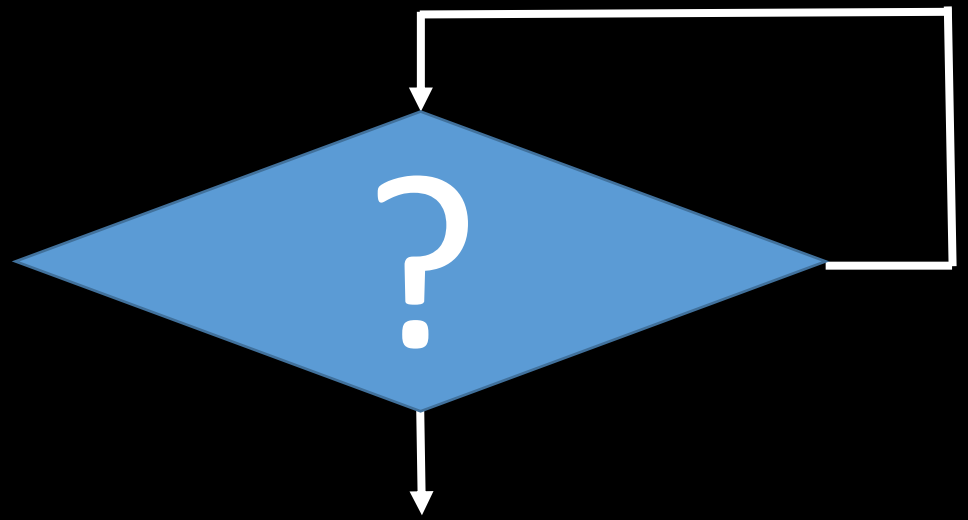
SELECTION STRUCTURE

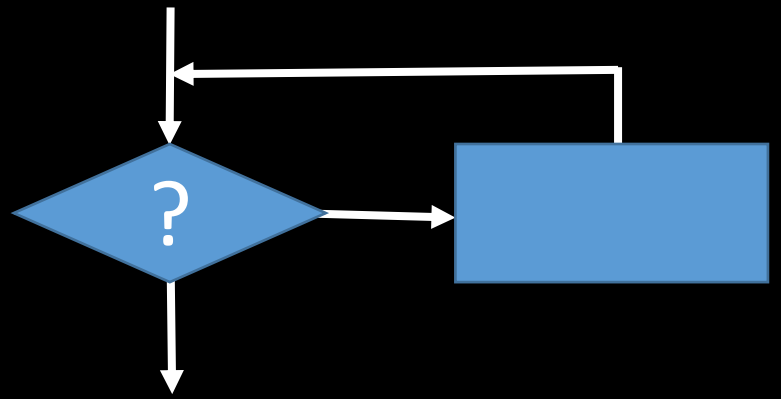
- In the flowchart segment below, the question “is x < y?” is asked
- If the answer is NO, then process A is performed
- If the answer is YES, then process B is performed



Iteration Structure

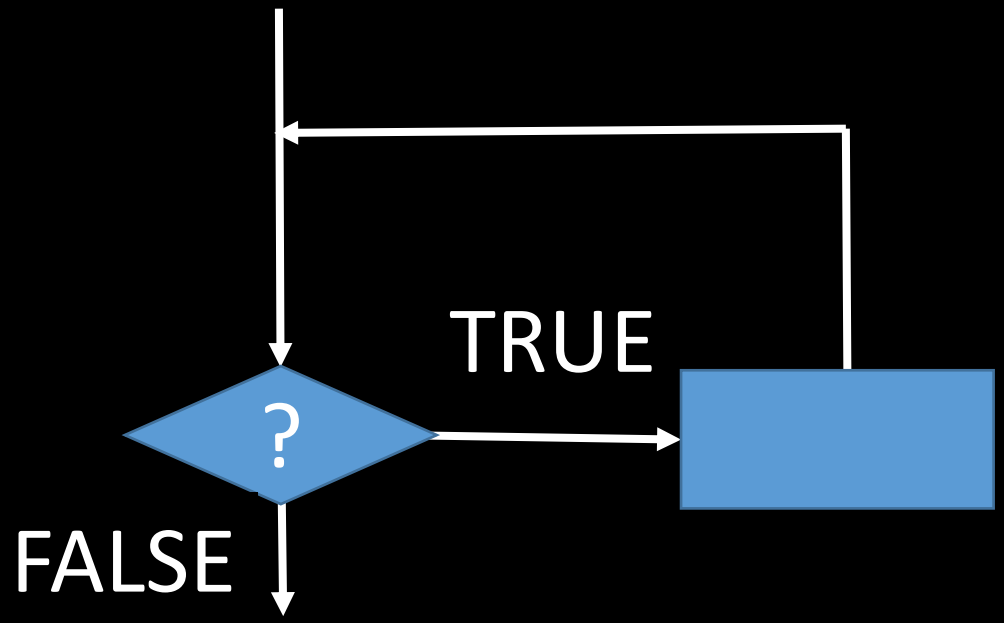
- An iteration structure represents part of the program that repeats
- This type of structure is commonly known as a loop





ITERATION STRUCTURE

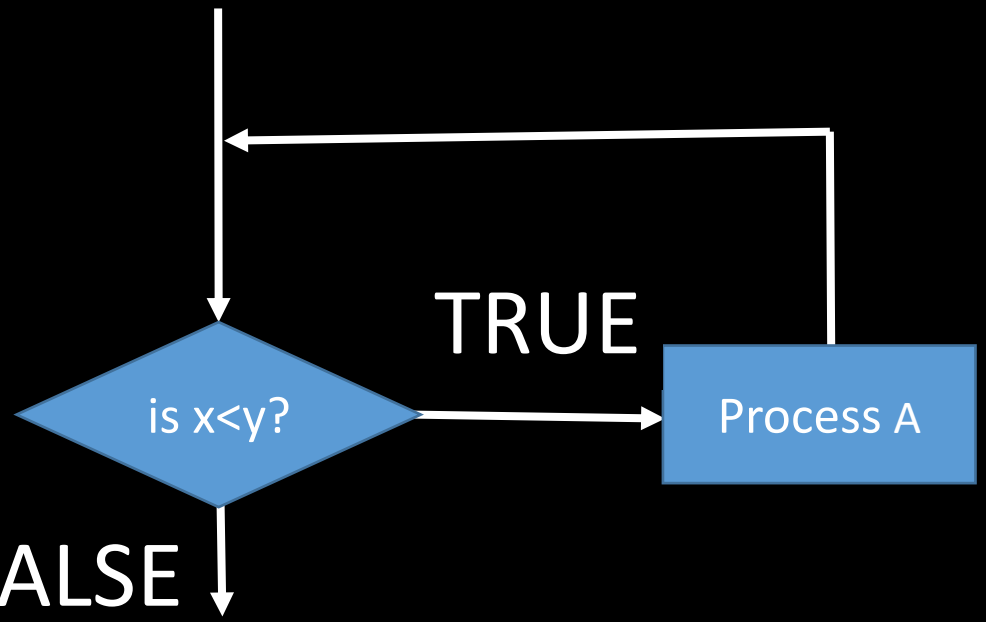
- A loop tests a condition, and if the condition exists, it performs an action
- Then it tests the condition again
- If the condition still exists, the action is repeated
- This continues until the condition no longer exists.



ITERATION STRUCTURE

- In the flowchart segment, the question “is x < y?” is asked
- If the answer is yes, then Process A is performed
- Then the question “is x < y?” is asked again
- Process A is repeated as long as x is less than y
- When x is no longer less than y, the iteration stops and the structure is exited

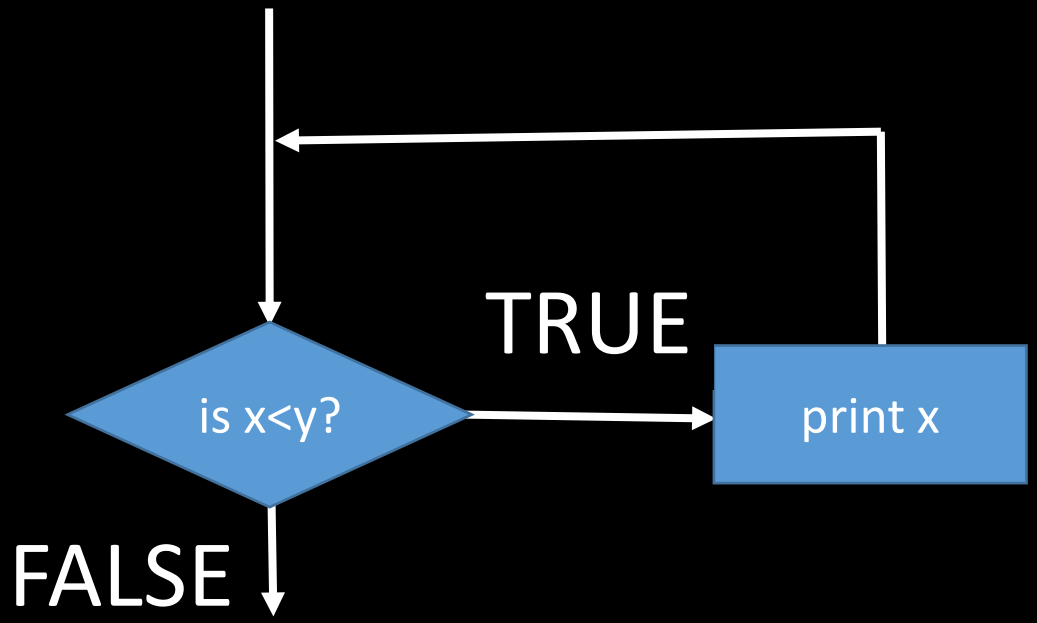
What needs to happen in Process A?



CONTROLLING AN ITERATION STRUCTURE

- The action performed by an iteration structure must eventually cause the loop to terminate. Otherwise, an infinite loop is created
- In this flowchart segment, x is never changed. Once the loop starts, it will never end.

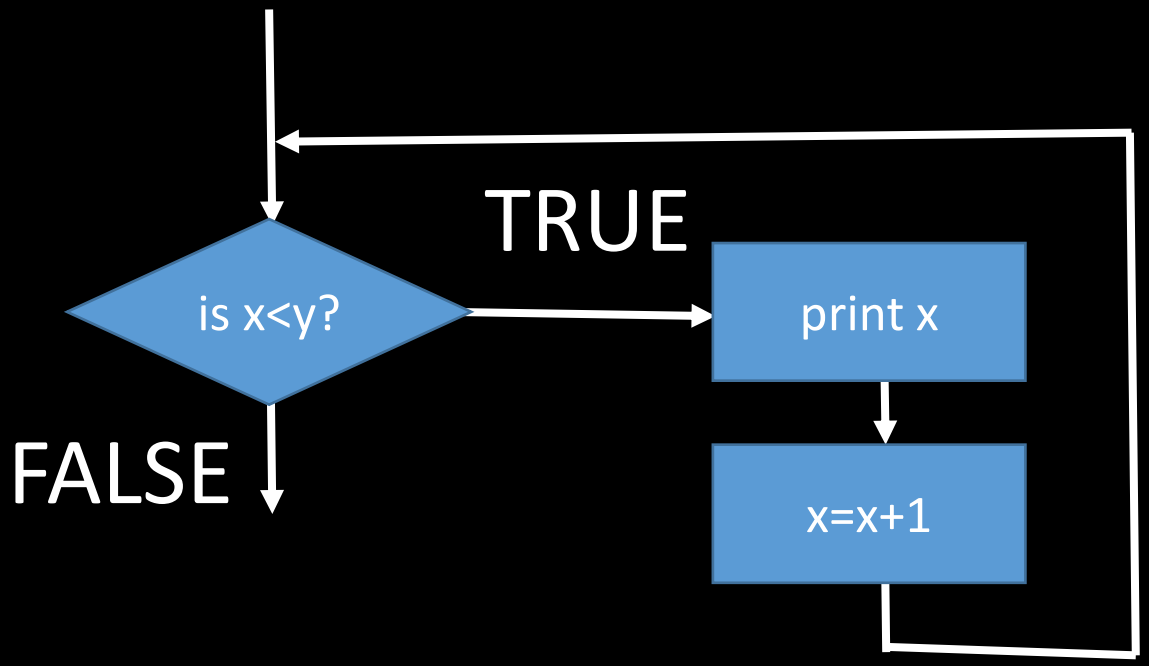
QUESTION: How can this flowchart be modified so it is no longer an infinite loop?



CONTROLLING AN ITERATION STRUCTURE

QUESTION: How can this flowchart be modified so it is no longer an infinite loop?

ANSWER: By adding an action within the iteration that changes the value of x



IF X STARTS AT 5 AND Y IS = 10?

How many times does the loop loop?

Print 9

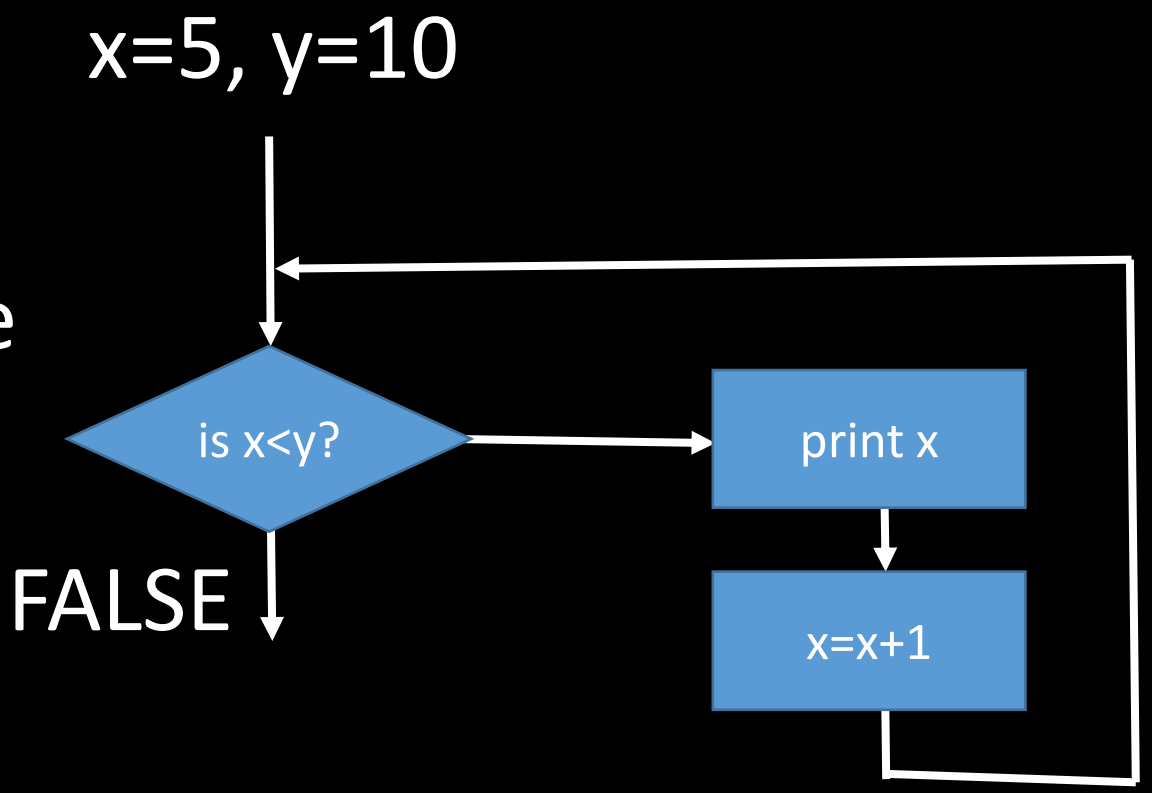
What is the value of y?

y = 10, and does not change

What is the value of x?

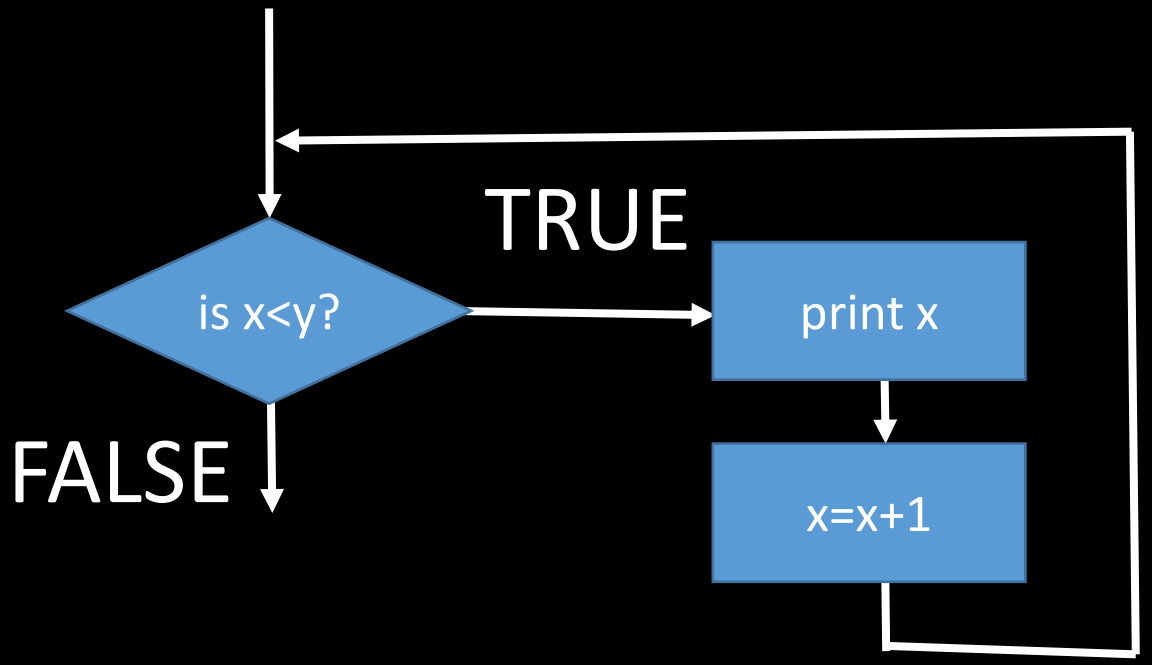
6^{th} test of x < y

x= 10



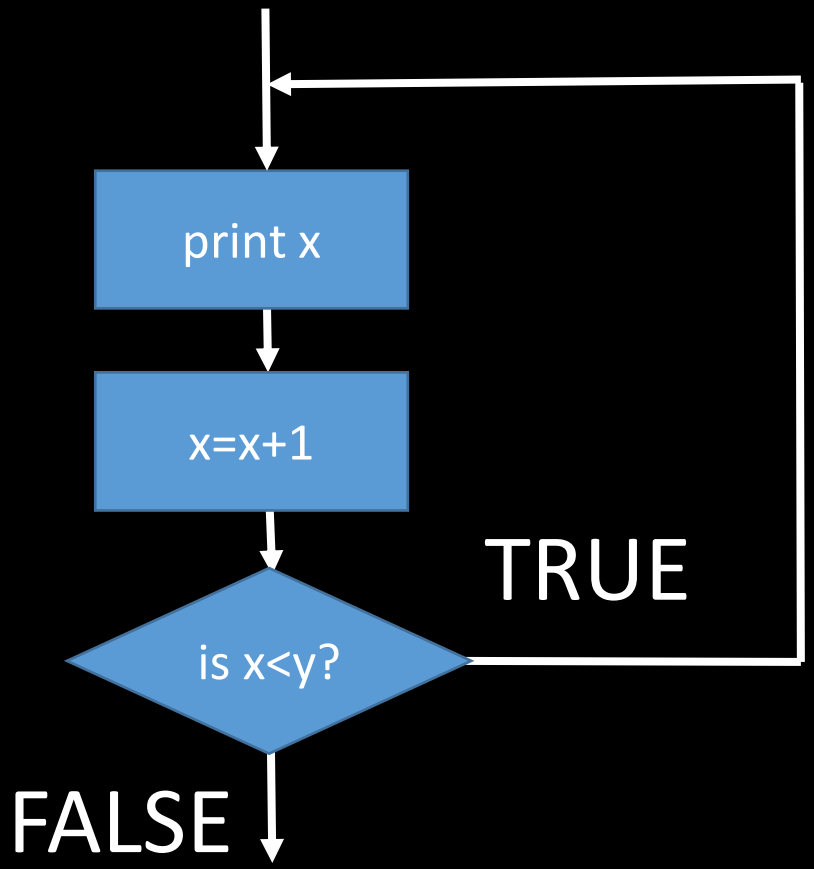
A PRE-TEST ITERATION STRUCTURE

- This type of structure is known as a pre-test iteration structure
- The condition is tested *BEFORE* any actions are performed
- In a pre-test iteration structure, if the condition is not TRUE, the loop will never begin



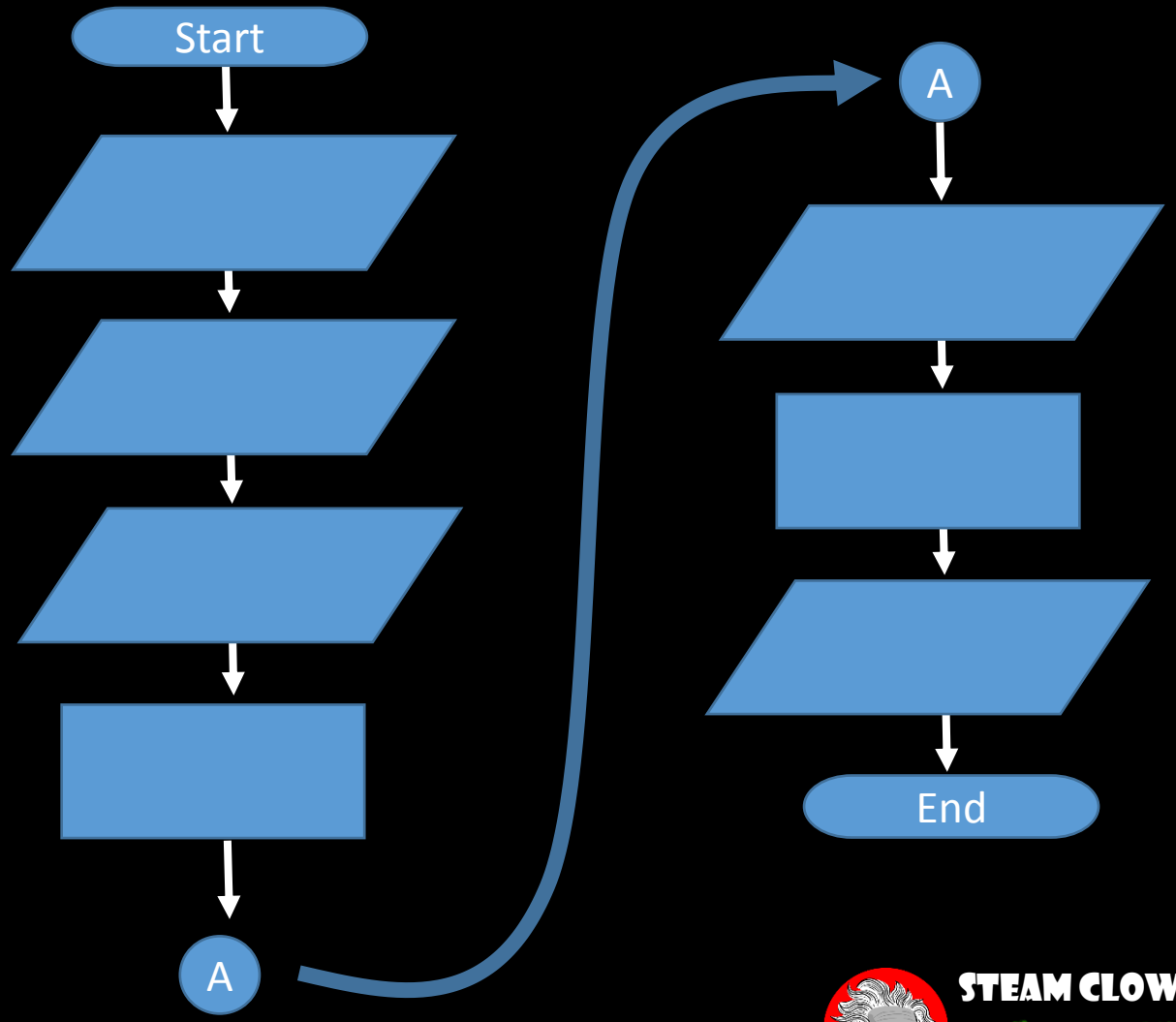
A POST-TEST ITERATION STRUCTURE

- This flowchart segment shows a post-test iteration structure
- The condition is tested AFTER the actions are performed
- A post-test iteration structure always performs its actions at least once



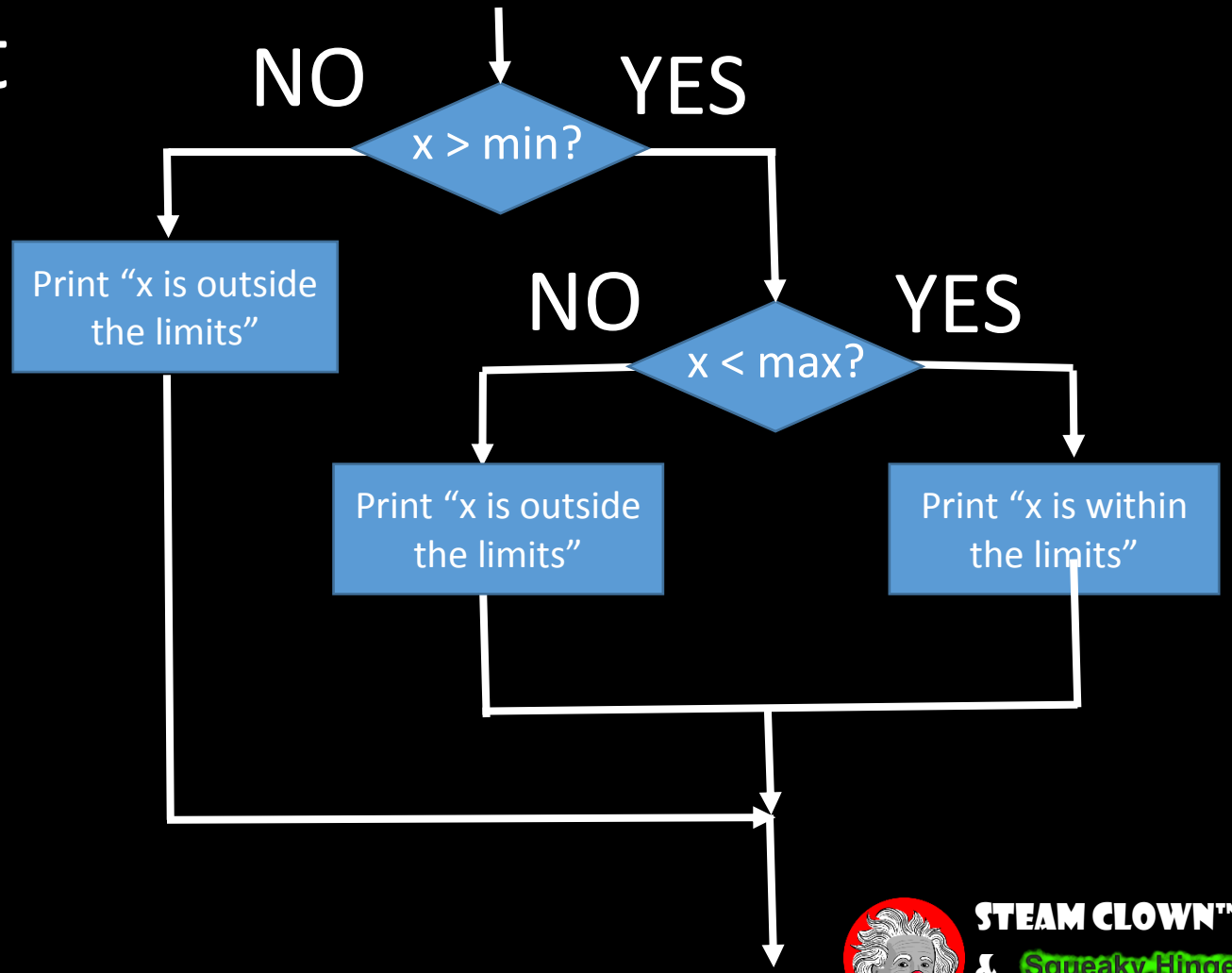
What if the flowchart can't fit on one page?

- A connector structure, lets you connect different flowchart segments
- The "A" connector indicates that the second flowchart segment begins where the first segment ends



COMBINING STRUCTURE

- This flowchart segment shows two selection structures combined



REFERENCE SLIDES

STEAM CLOWN™
& Squeaky Hinge
PRODUCTIONS

STEAM CLOWN™ PRODUCTIONS

APPENDIX

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

Appendix A: License & Attribution

- This interpretation is primarily the Intellectual Property of Jim Burnham, Top STEAM Clown, at STEAMClown.org
- This presentation and content is distributed under the Creative Commons License CC-by-nc-sa-3.0
- My best attempt to properly attribute, or reference any other images, sources or work I have used are listed in Appendix B

Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial — You may not use the material for commercial purposes.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS

Appendix B: Attribution for Sources Used

- This tutorial is based upon Appendix C from "*Starting Out with C++: From Control Structures to Objects* (5th Edition)" - Copyright Tony Gaddis 2007, Published by Addison-Wesley

STEAM CLOWN™ & Squeaky Hinge PRODUCTIONS