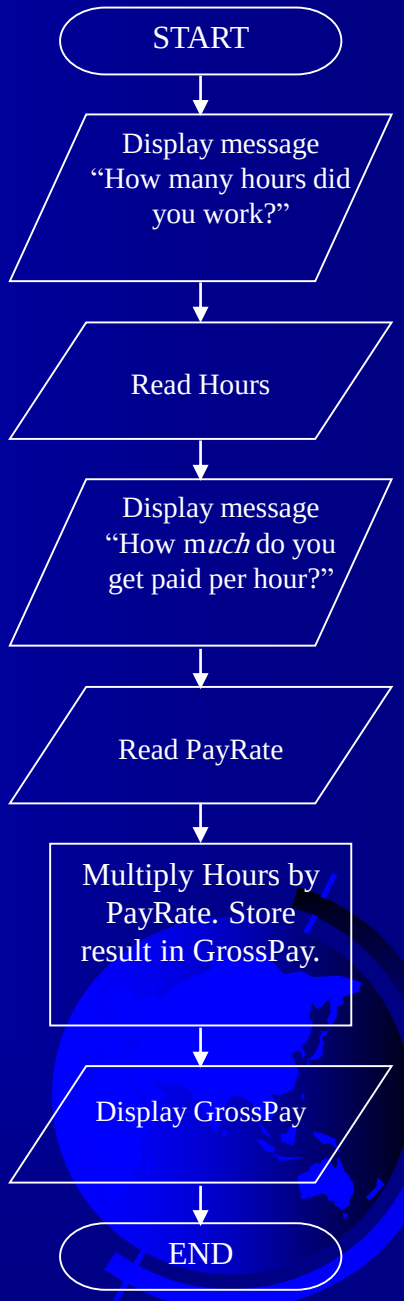
COCS 202 Programming I

Chapter 1 Introduction to Flowcharts

Hossam M. J. Mustafa
FCITR, KAUR

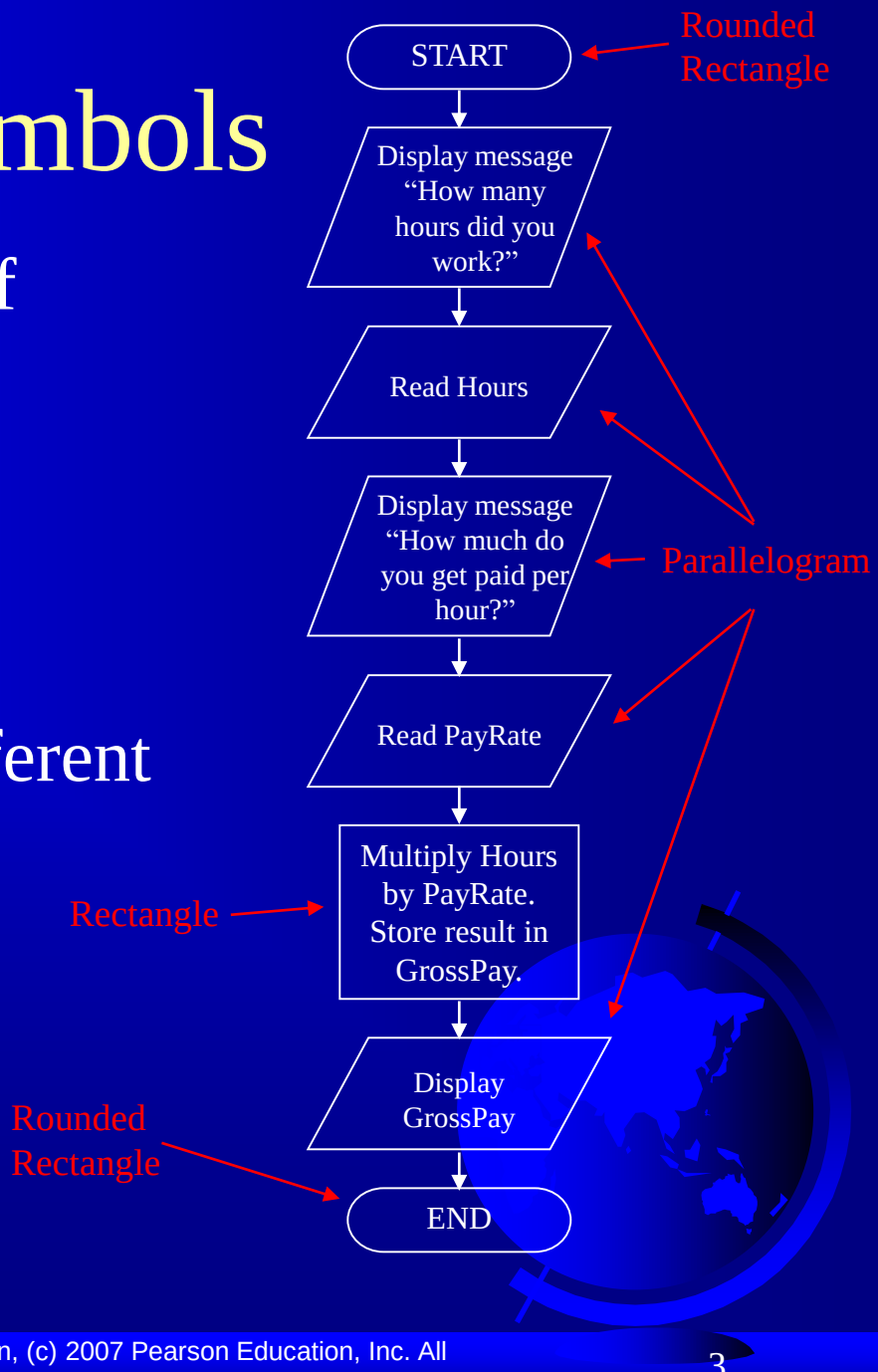
What is a Flowchart?

- 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

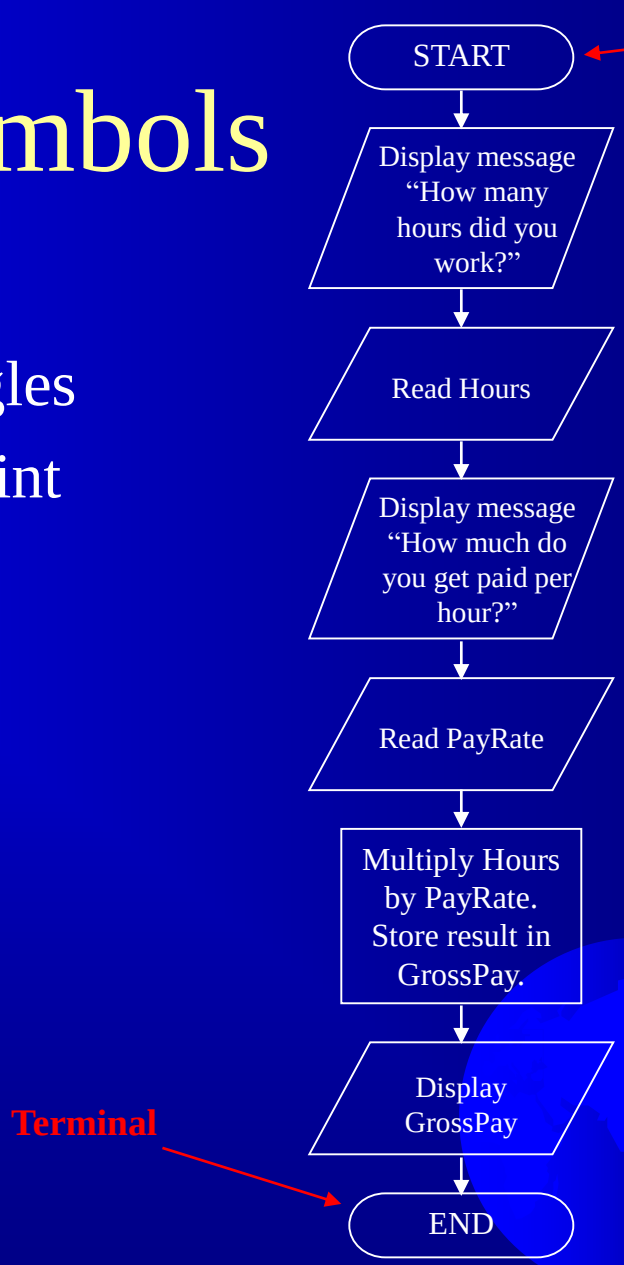


Basic Flowchart Symbols

- Terminals
  - represented by rounded rectangles
  - indicate a starting or ending point

START

END

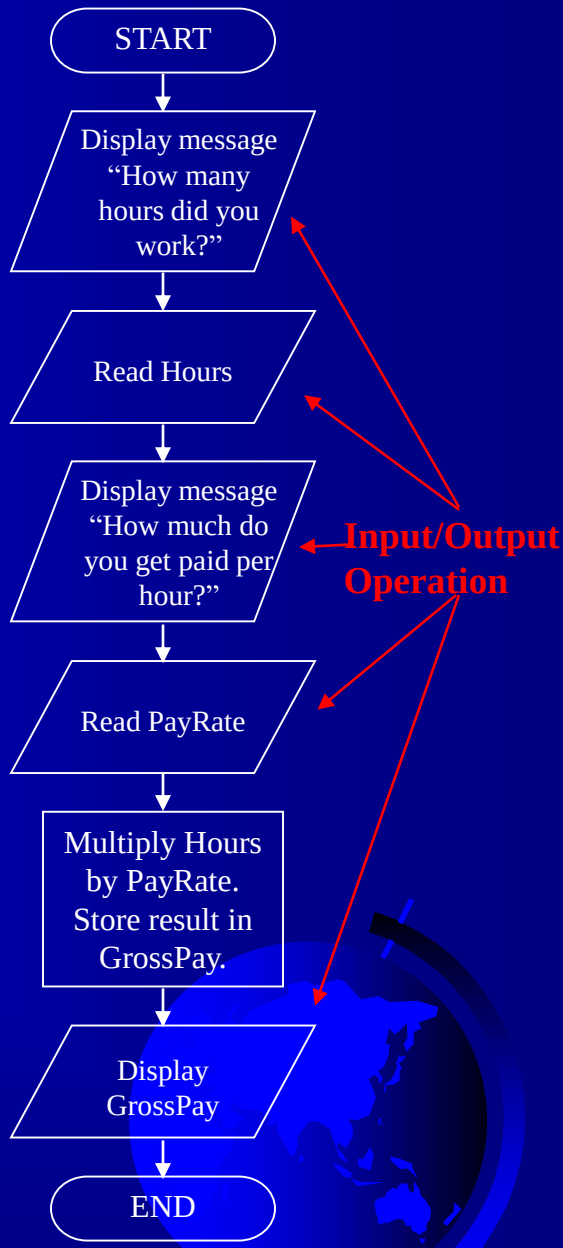


Basic Flowchart Symbols

- Input/Output Operations
  - represented by parallelograms
  - indicate an input or output operation

Display message "How many hours did you work?"

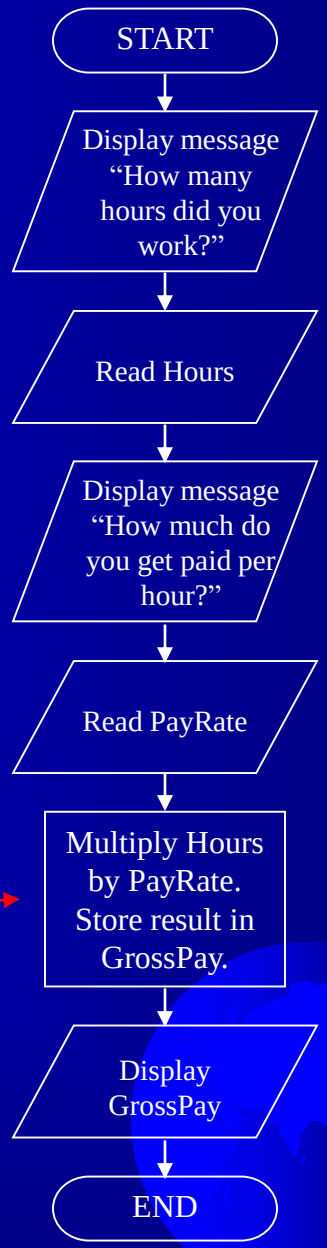
Read Hours



Basic Flowchart Symbols

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

Multiply Hours by PayRate. Store result in GrossPay.



Process

Stepping Through the Flowchart

- In the next slides we will step through each symbol in the flowchart. We will show the program output and the contents of the variables.

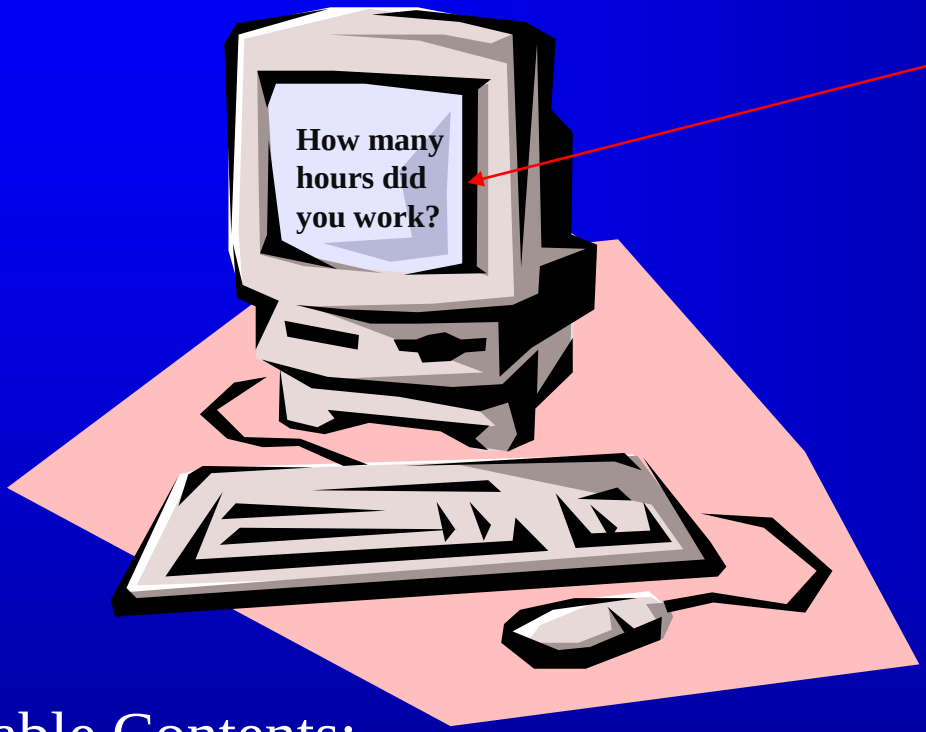
- Variable Contents:

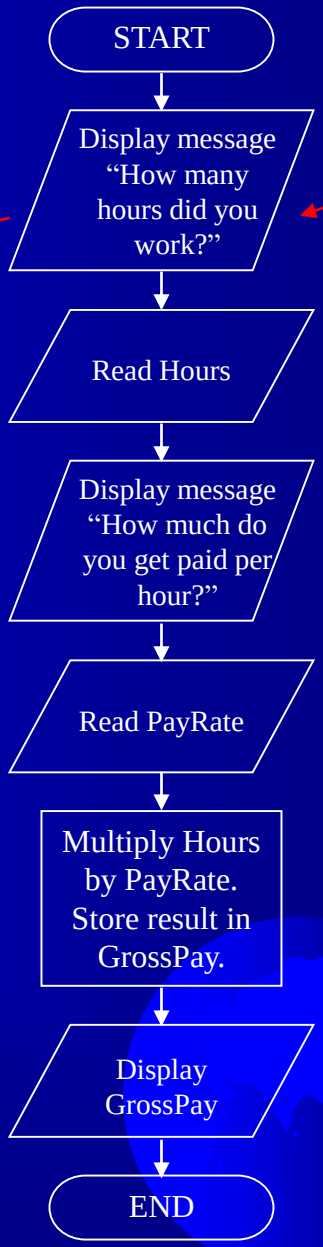
  Hours: ?

  PayRate: ?

  GrossPay: ?

Stepping Through the Flowchart





Step 1: An Output Operation

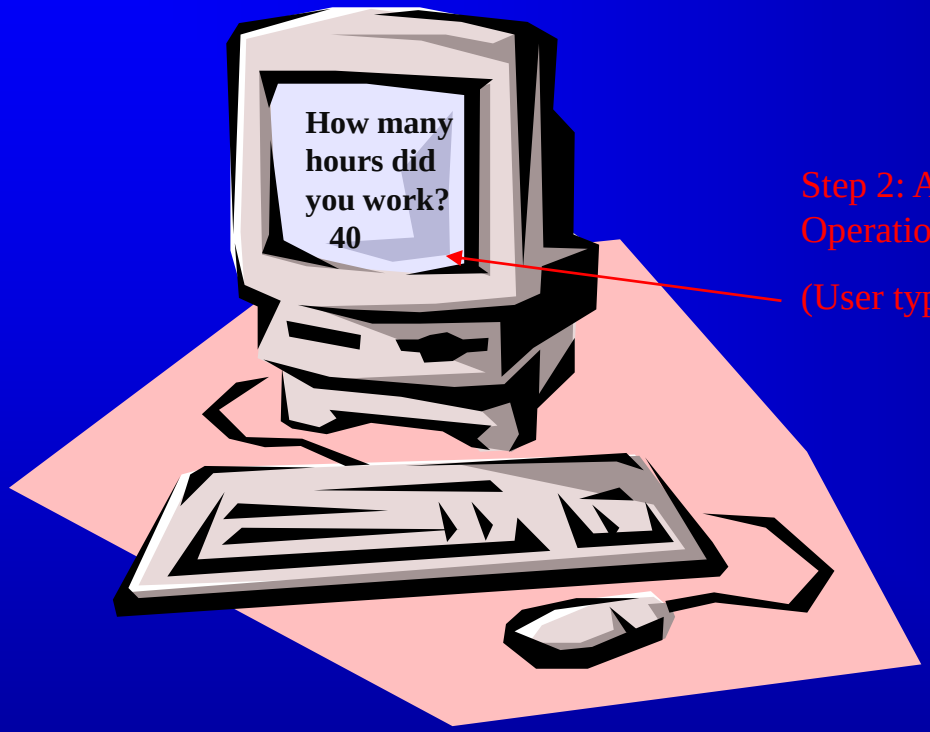
Variable Contents:

Hours: ?

PayRate: ?

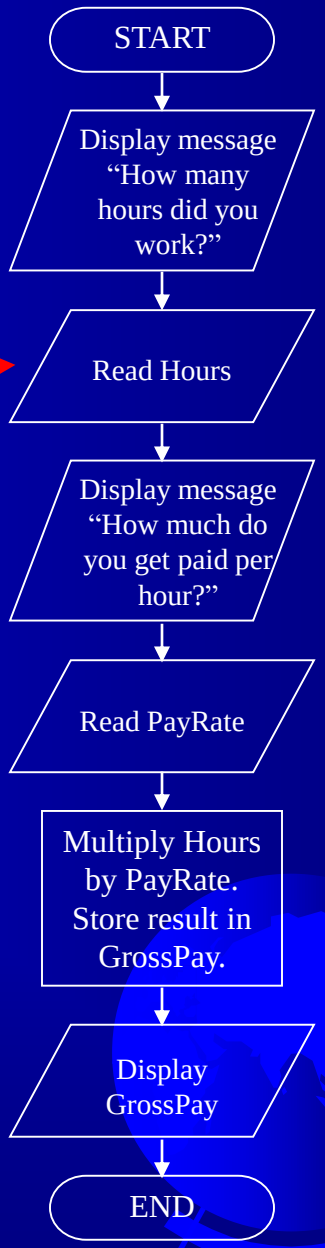
GrossPay: ?

Stepping Through the Flowchart



Step 2: An Input Operation

(User types 40)



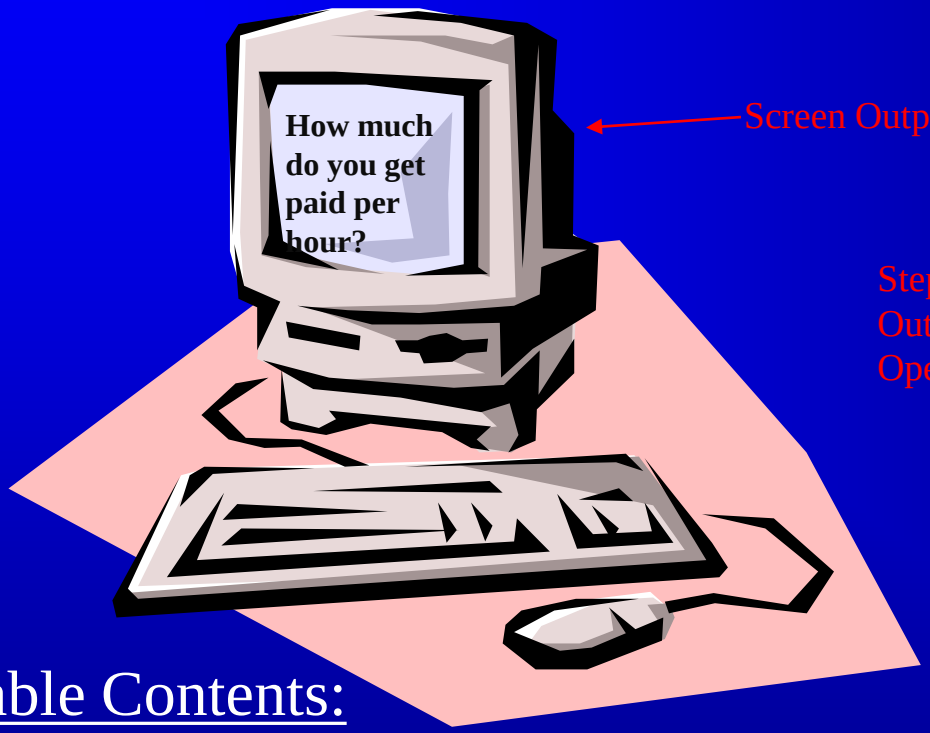
Variable Contents:

Hours: 40 ← The value 40 is stored in Hours.

PayRate: ?

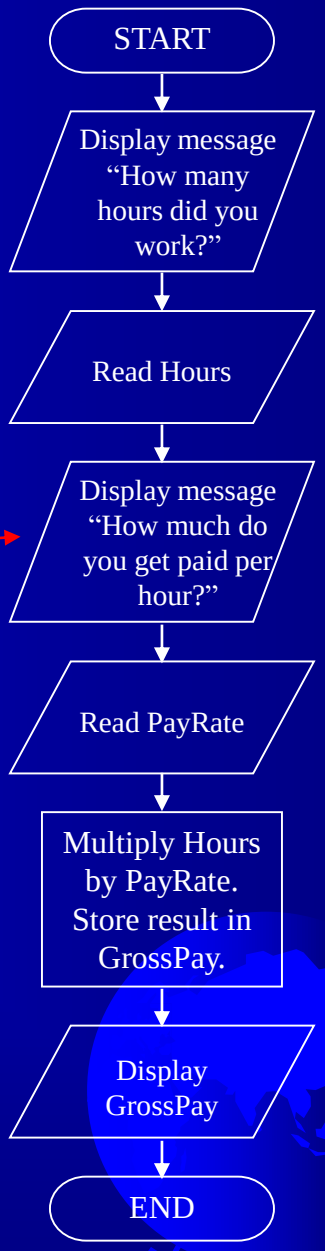
GrossPay: ?

Stepping Through the Flowchart



Screen Output

Step 3: An Output Operation



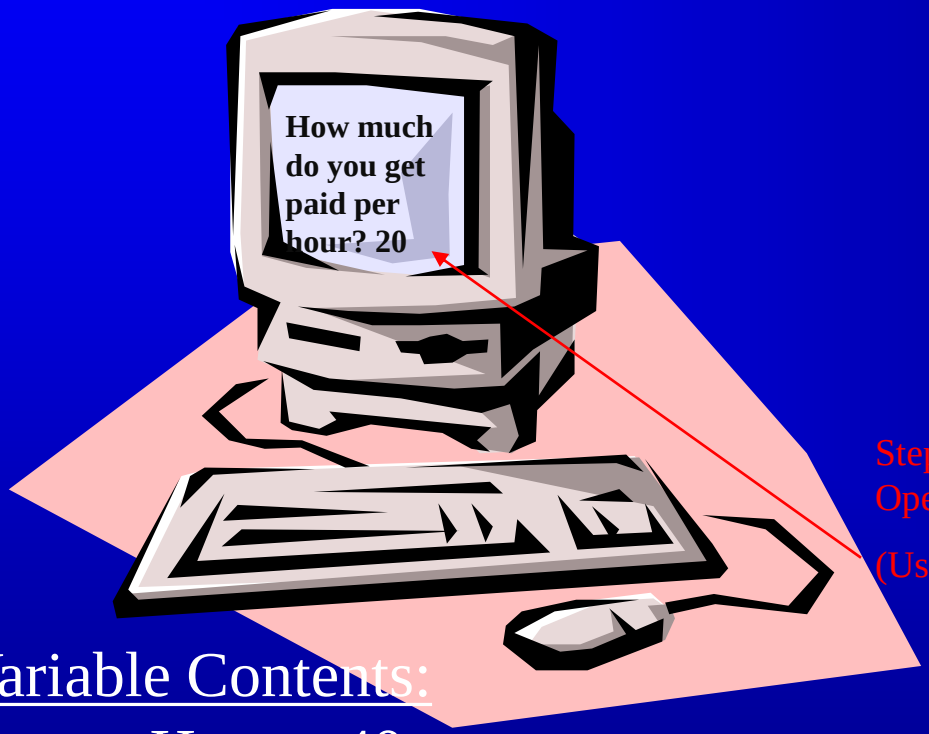
Variable Contents:

Hours: 40

PayRate: ?

GrossPay: ?

Stepping Through the Flowchart



Step 4: Input Operation

(User types 20)

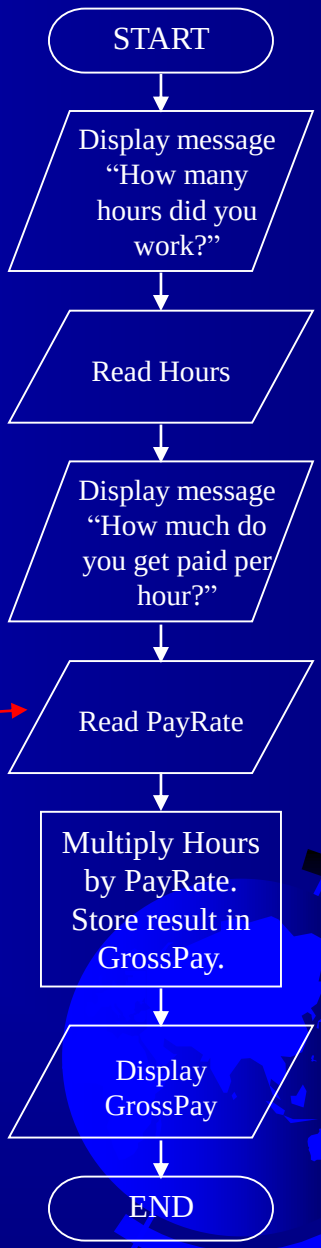
Variable Contents:

Hours: 40

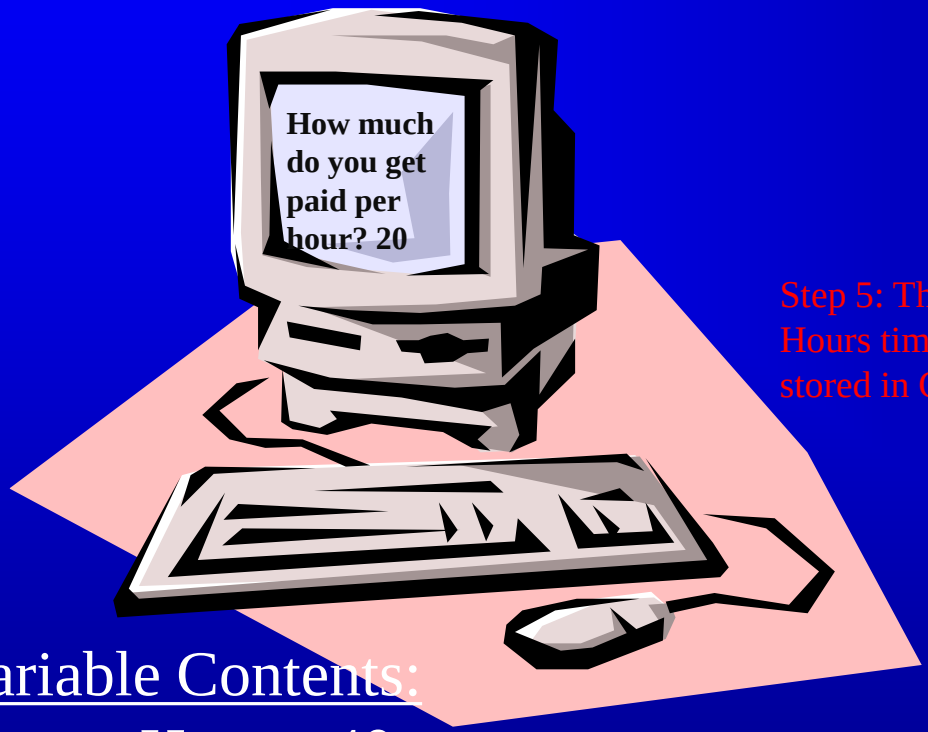
PayRate: 20

GrossPay: ?

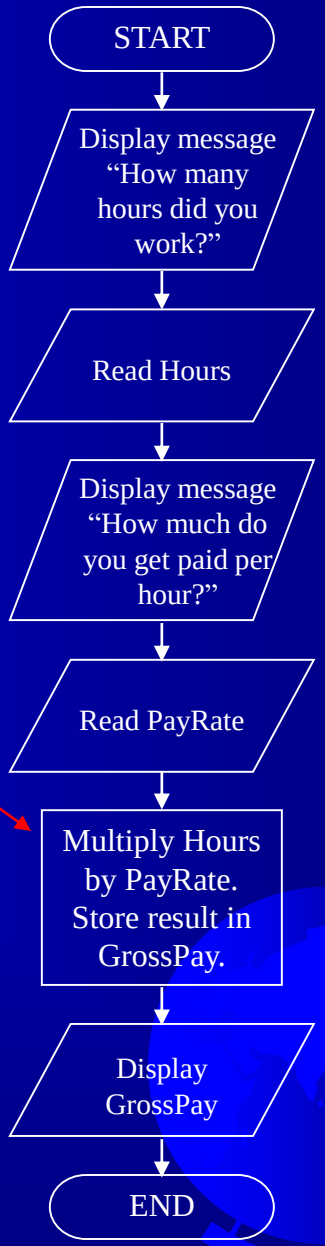
The value 20 is stored in PayRate.



Stepping Through the Flowchart



Step 5: The product of Hours times PayRate is stored in GrossPay



Variable Contents:

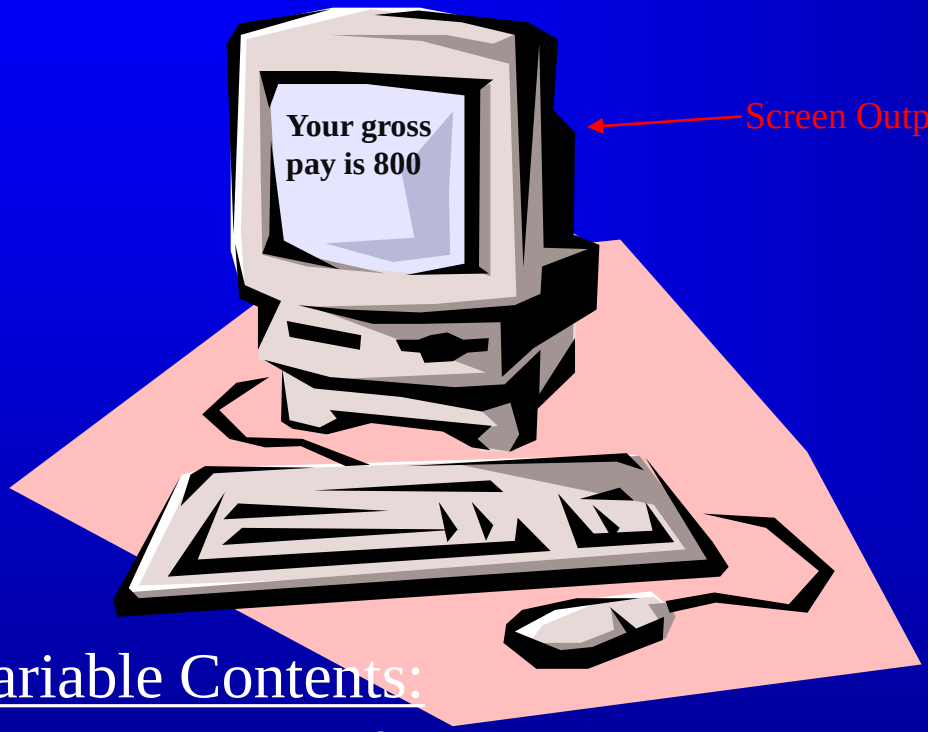
Hours: 40

PayRate: 20

GrossPay: 800

The value 800 is stored in GrossPay.

Stepping Through the Flowchart



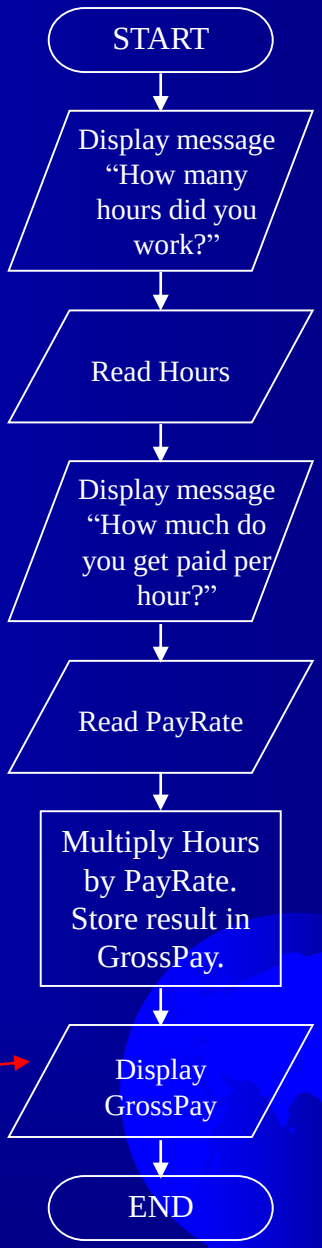
Screen Output

Variable Contents:

Hours: 40

PayRate: 20

GrossPay: 800



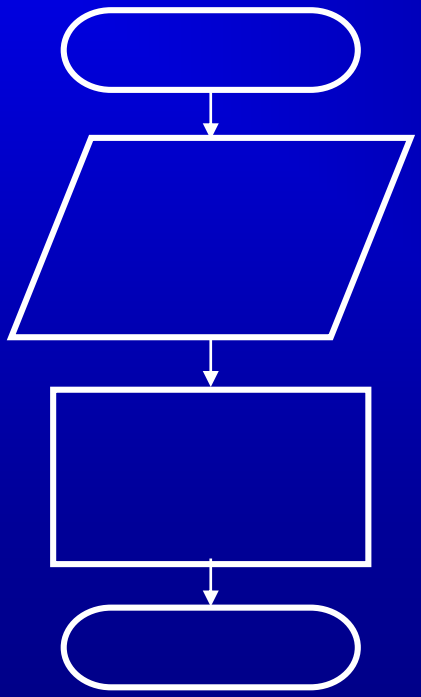
Step 6: An Output Operation

Three Flowchart Structures

- Sequence
- Selection
- Iteration

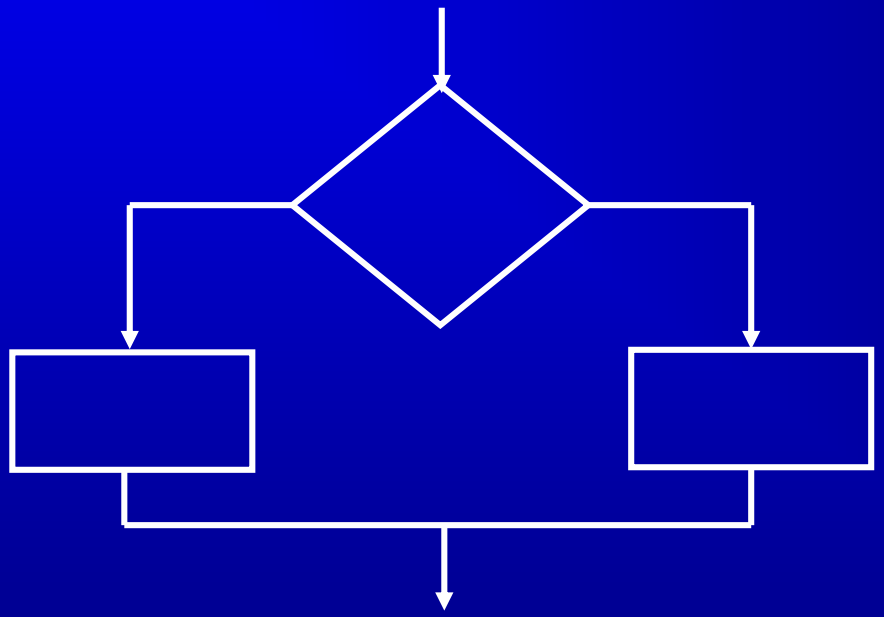
Sequence Structure

- A series of actions are performed in sequence
- The pay-calculating example was a sequence flowchart.



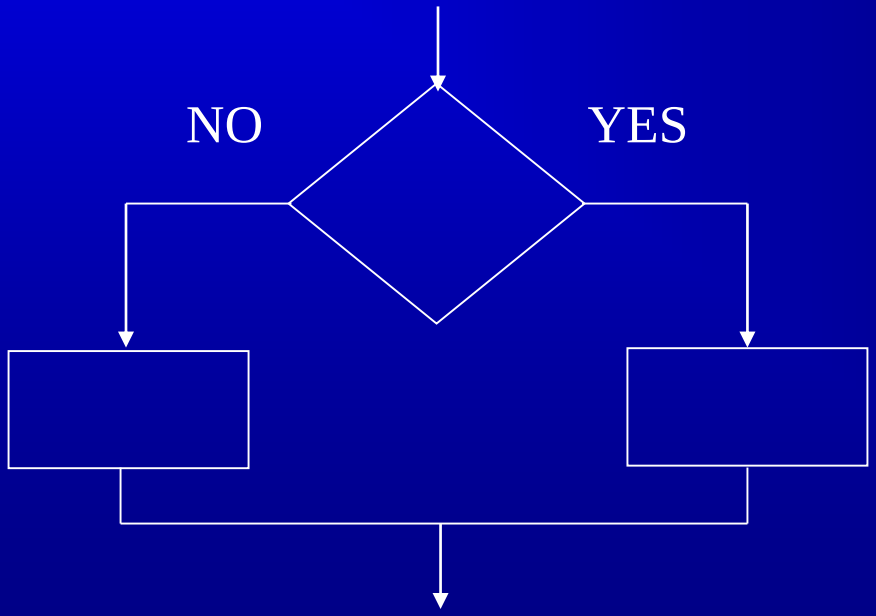
Selection Structure

- One of two possible actions is taken, depending on a condition.



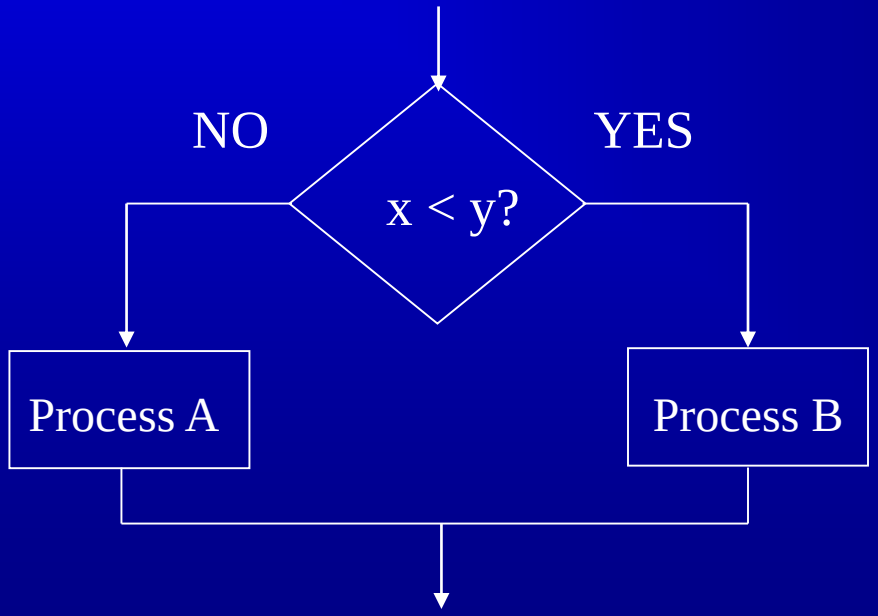
Selection Structure

- 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

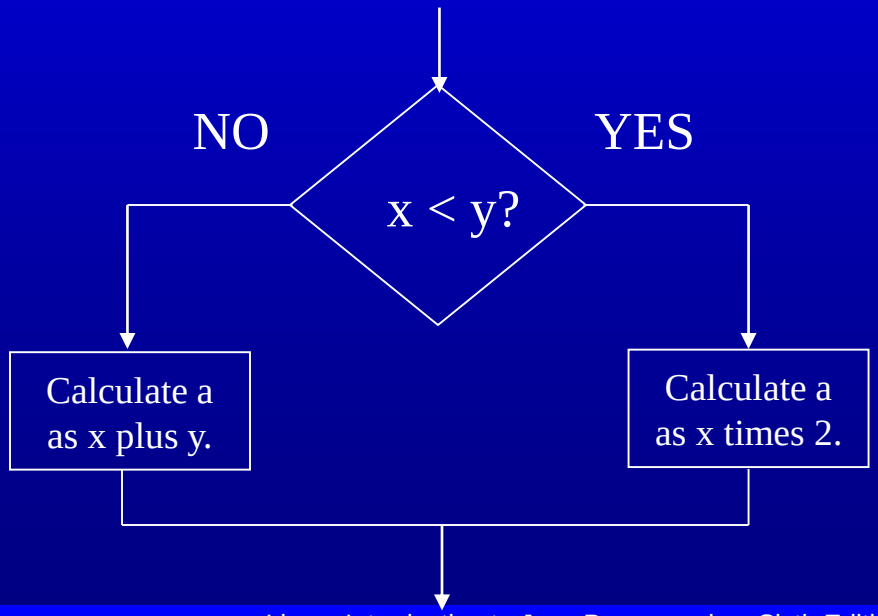
- In the flowchart segment below, the question "is x < y?" is asked. If the answer is no, then process A is performed. If the answer is yes, then process B is performed.



Selection Structure

- The flowchart segment below shows how a decision structure is expressed in Java as an if/else statement.

Flowchart



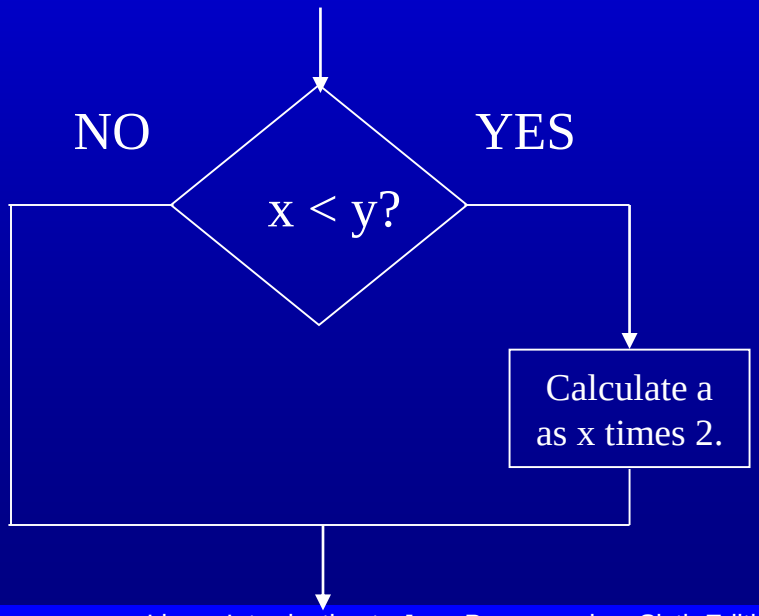
Java Code

```
if (x < y)
    a = x * 2;
else
    a = x + y;
```

Selection Structure

- The flowchart segment below shows a decision structure with only one action to perform. It is expressed as an if statement in Java code.

Flowchart

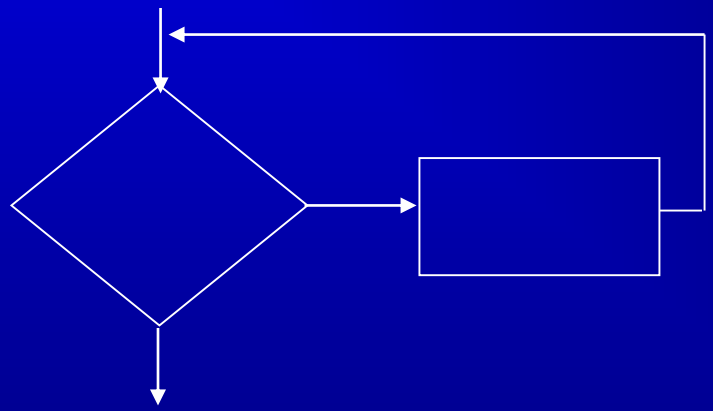


Java Code

```
if (x < y)
    a = x * 2;
```

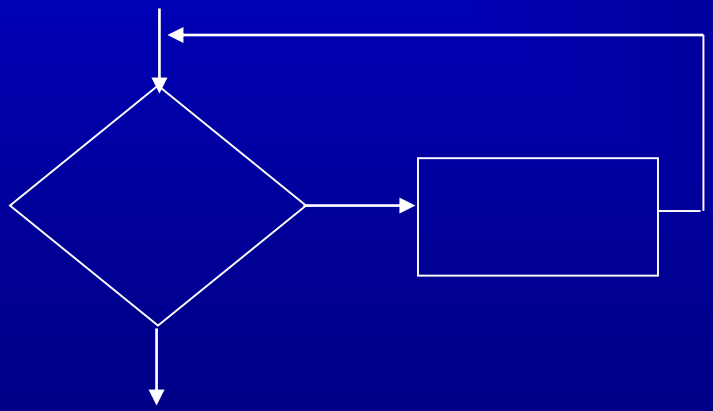
Iteration Structure

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



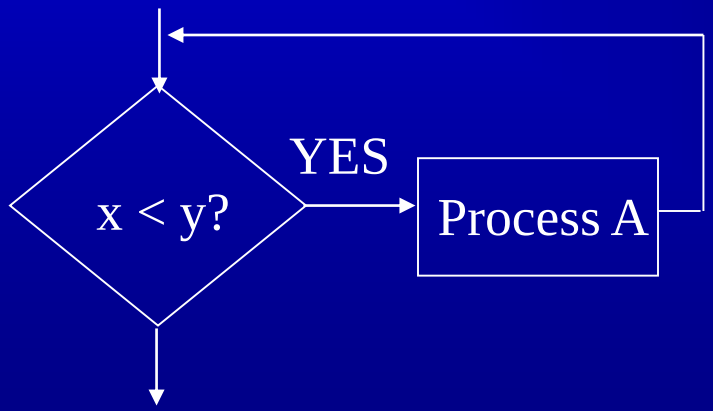
Iteration Structure

- Notice the use of the diamond symbol. 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

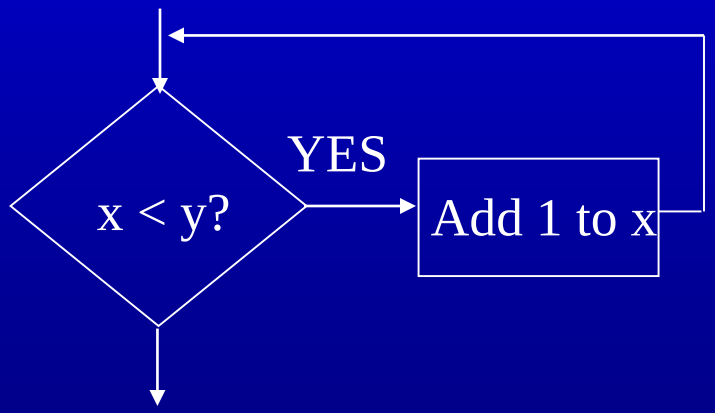
- In the flowchart segment, the question "is x < y?" is asked. If the answer is yes, then Process A is performed. The question "is x < y?" is asked again. Process A is repeated as long as x is less than y. When x is no longer less than y, the iteration stops and the structure is exited.



Iteration Structure

- The flowchart segment below shows an iteration structure expressed in Java as a while loop.

Flowchart

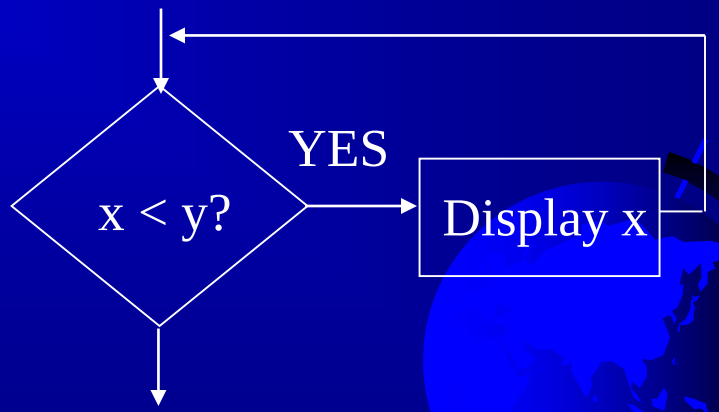


Java Code

```
while (x < y)
  x++;
```

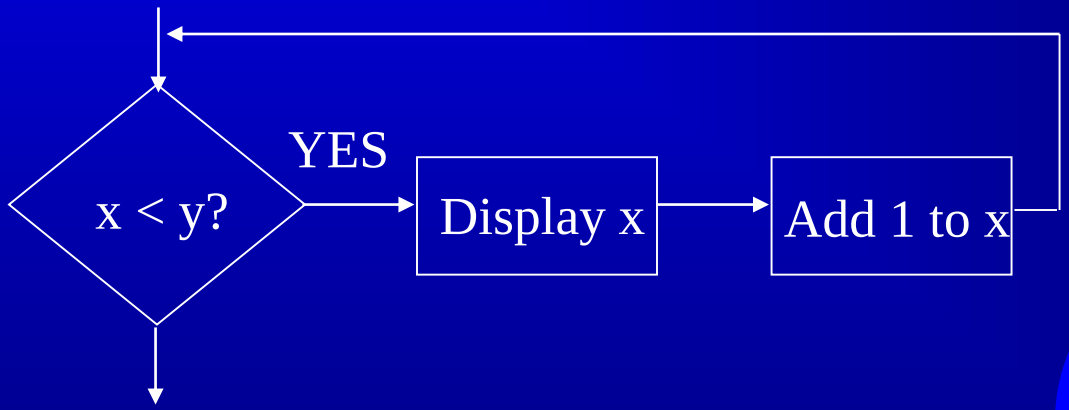
Controlling an Iteration Structure

- 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?



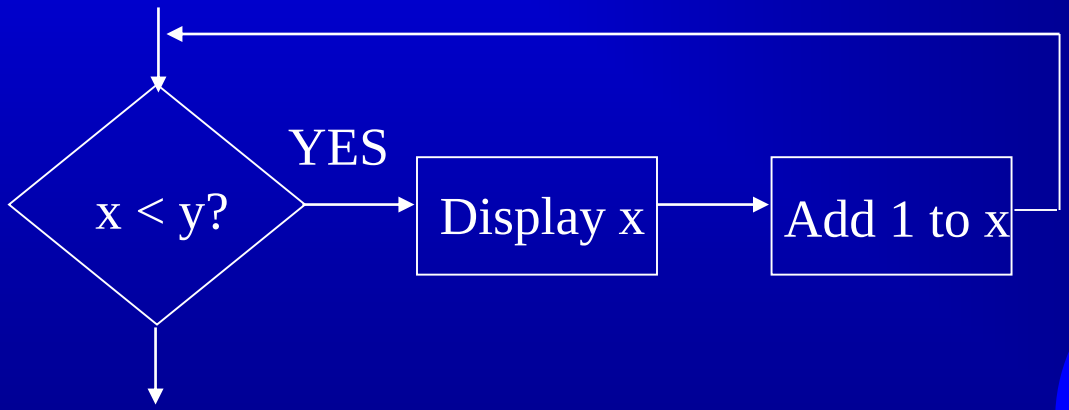
Controlling an Iteration Structure

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



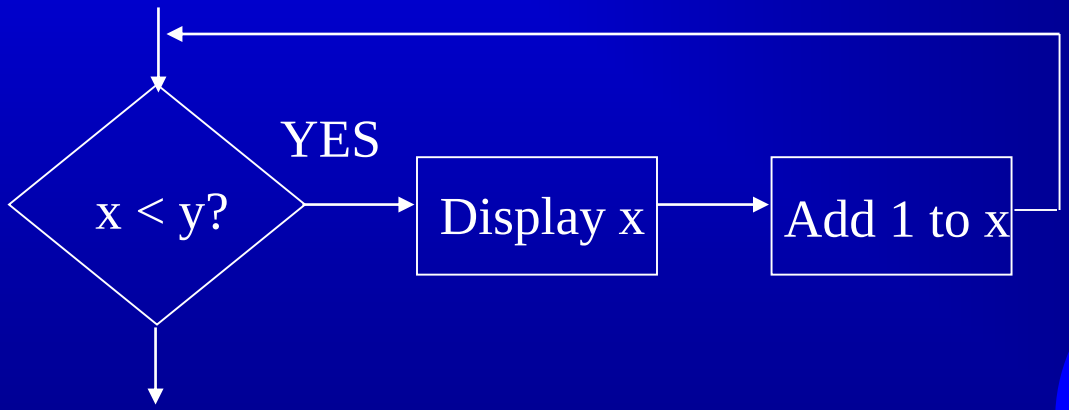
A Pre-Test Iteration Structure

- This type of structure is known as a pre-test iteration structure. The condition is tested *BEFORE* any actions are performed.



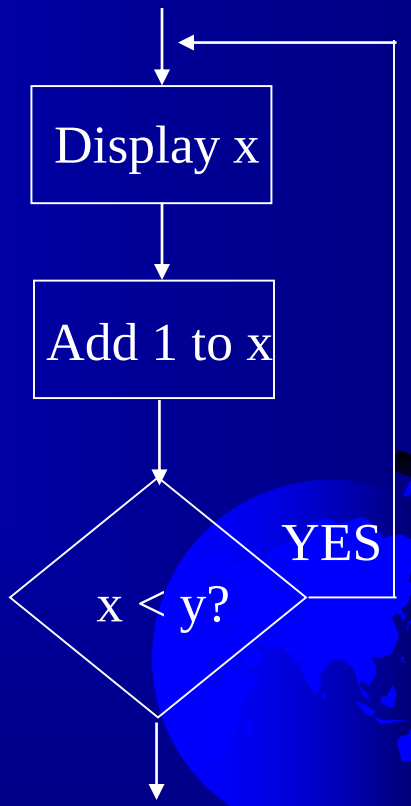
A Pre-Test Iteration Structure

- In a pre-test iteration structure, if the condition does not exist, the loop will never begin.



A Post-Test Iteration Structure

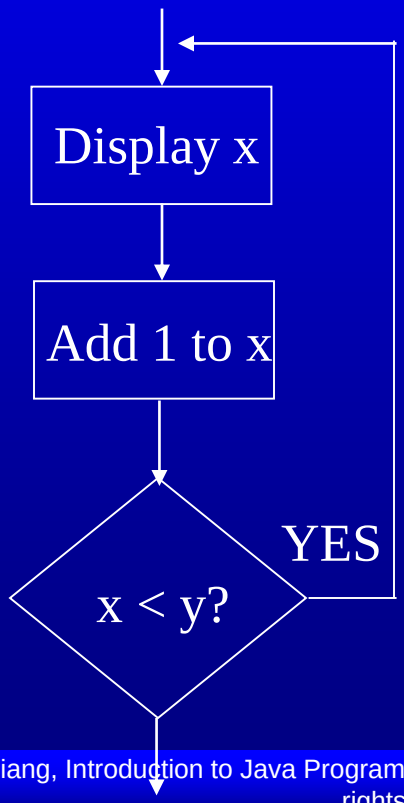
- This flowchart segment shows a post-test iteration structure.
- The condition is tested *AFTER* the actions are performed.
- A post-test iteration structure always performs its actions at least once.



A Post-Test Iteration Structure

- The flowchart segment below shows a post-test iteration structure expressed in *Java* as a do-while loop.

Flowchart



Java Code

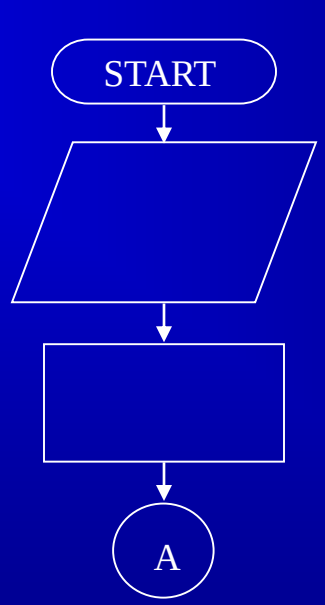
```
do
{
    cout << x << endl;
    x++;
} while (x < y);
```

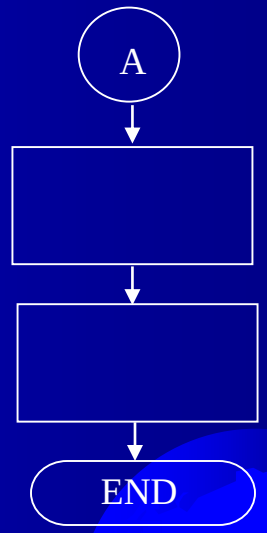
Connectors

- Sometimes a flowchart will not fit on one page.
- A connector (represented by a small circle) allows you to connect two flowchart segments.

Connectors

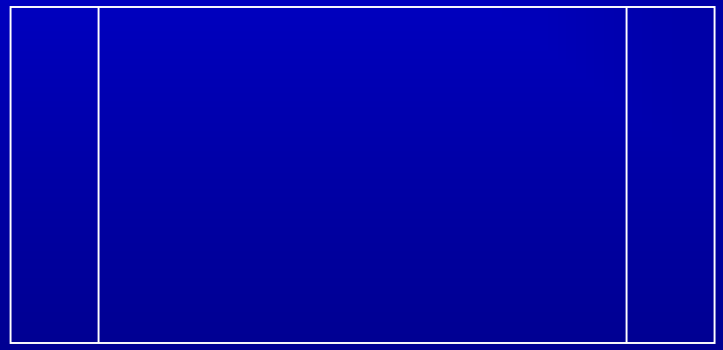
•The “A” connector indicates that the second flowchart segment begins where the first segment ends.





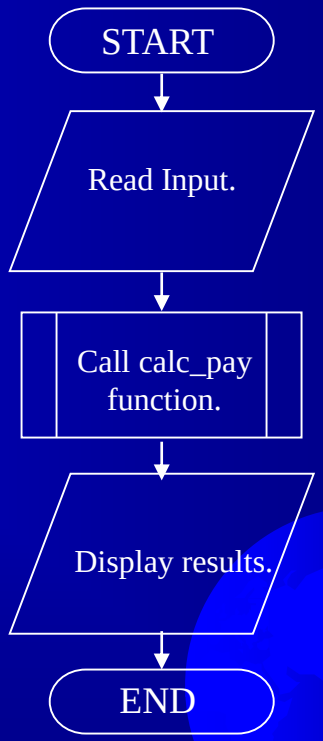
Modules

- A program module, such as a subprogram (or function in Java is represented by a special symbol.



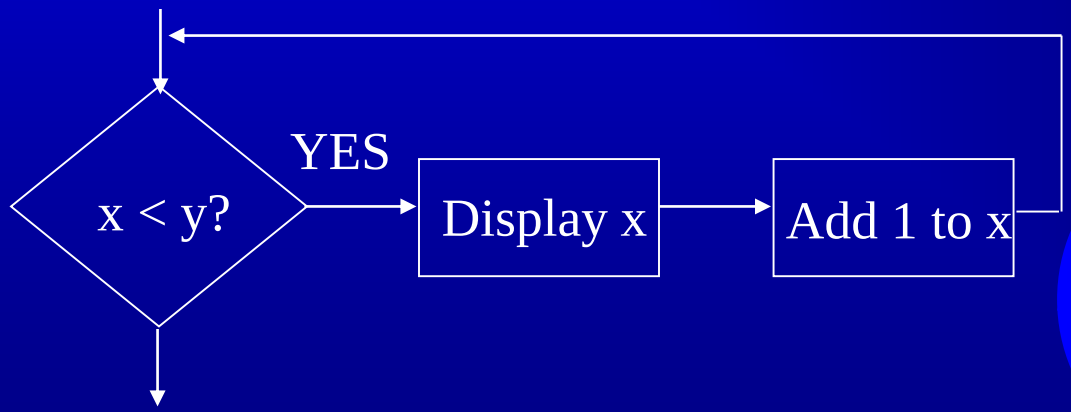
Modules

- The position of the module symbol indicates the point the module is executed.
- A separate flowchart can be constructed for the module.



Combining Structures

- Structures are commonly combined to create more complex algorithms.
- The flowchart segment below combines a selection structure with a sequence structure.



Combining Structures

- This flowchart segment shows two selection structures combined.