Objectives for class 5

--- Chapter 3---

3.5 To implement selection control with nested if and multi-way if-elif-else statements (§3.6).

3.6 To combine conditions using logical operators (and, or, and not) (§3.10).

3.7 To use selection statements with combined conditions (§§3.11–3.12).

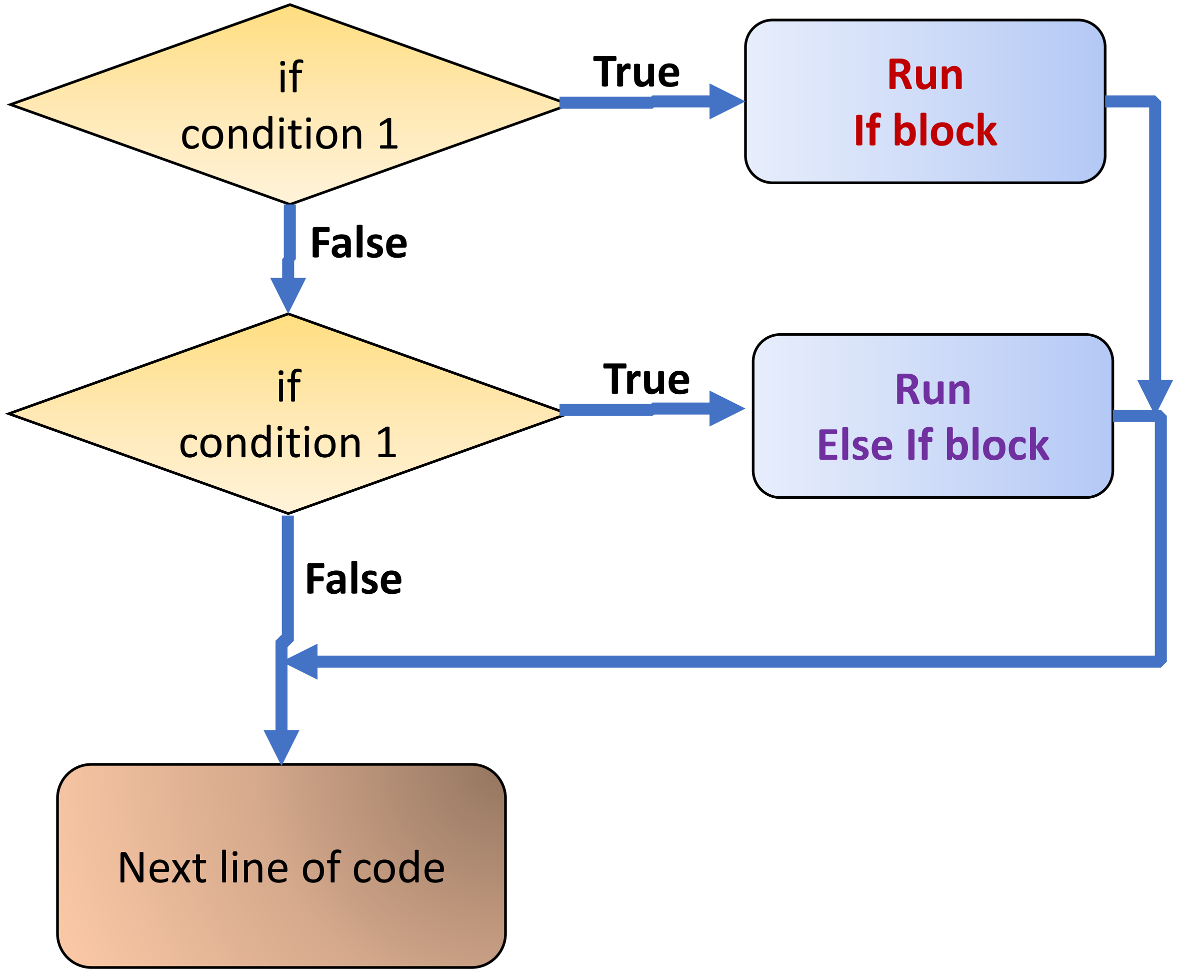
3.8 To understand how to develop a program with selections.

Review Assignment 1 and Assignment 2

Review Quiz 1

Multiple-way Decisions : select one block of many to execute (No ELSE)

```
if <condition 1> :
    <statement>
    <statement>
    <statement>
elif <condition 2> :
    <statement>
    <statement>
<Next line of code>
```

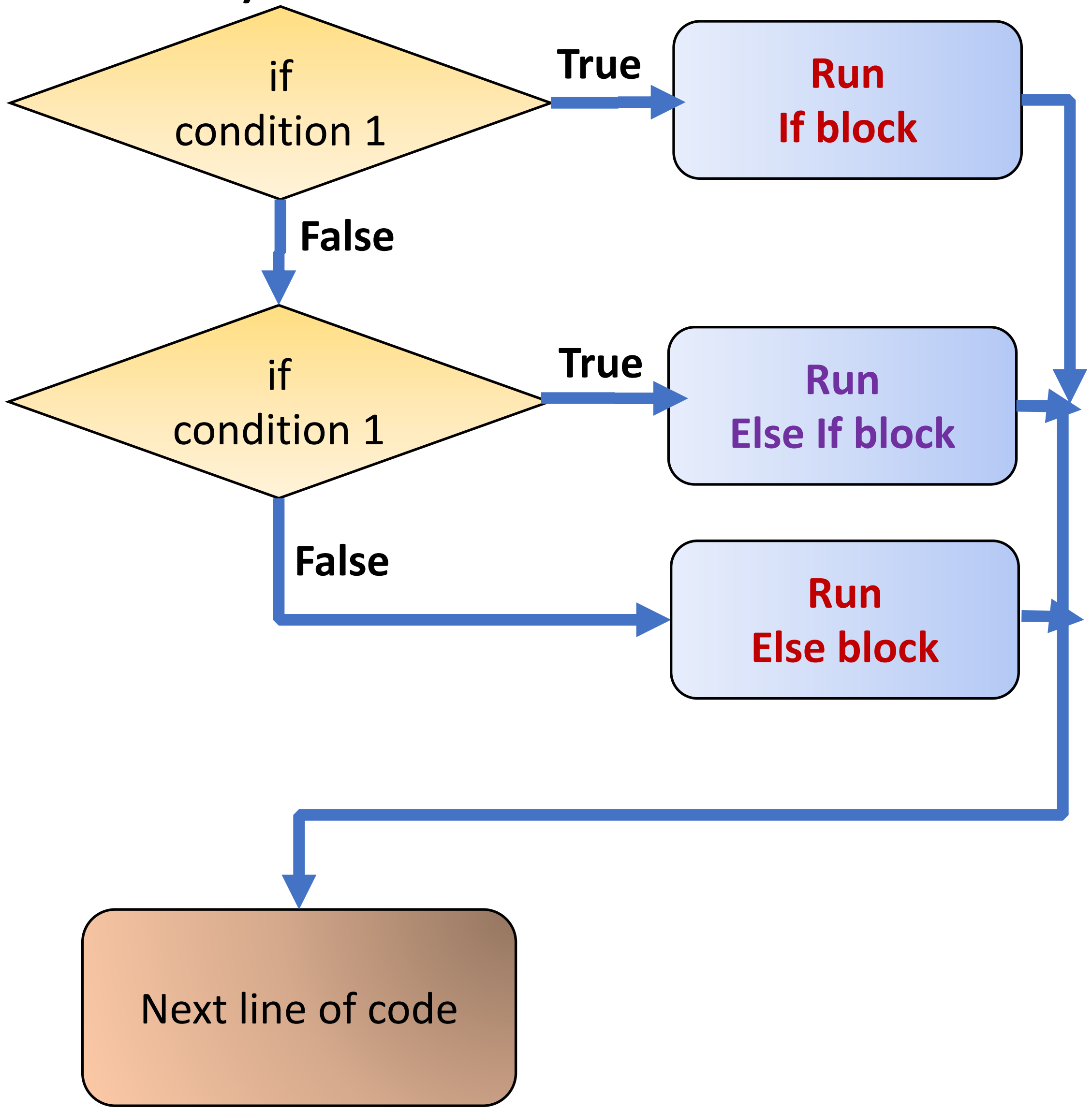


```
x=100

if x < 100:
    print(x, " is less than 100")
elif x > 100:
    print(x, " is greater than 100")

print("Done")
```

“Done”

Multiple-way Decisions : select one block of many to execute (With ELSE)

```
if <condition 1> :
    <statement>
    <statement>
    <statement>
elif <condition 2> :
    <statement>
    <statement>
else:
    <statement>
    <statement>
<Next line of code>
```



```
x=100

if x < 100:
    print(x, " is less than 100")
elif x > 100:
    print(x, " is greater than 100")
else:
    print(x, " is equal to 100")

print("Done")
```

“100 is equal to 100”

“Done”

Practice

Which message will never be printed regardless of the value for x?

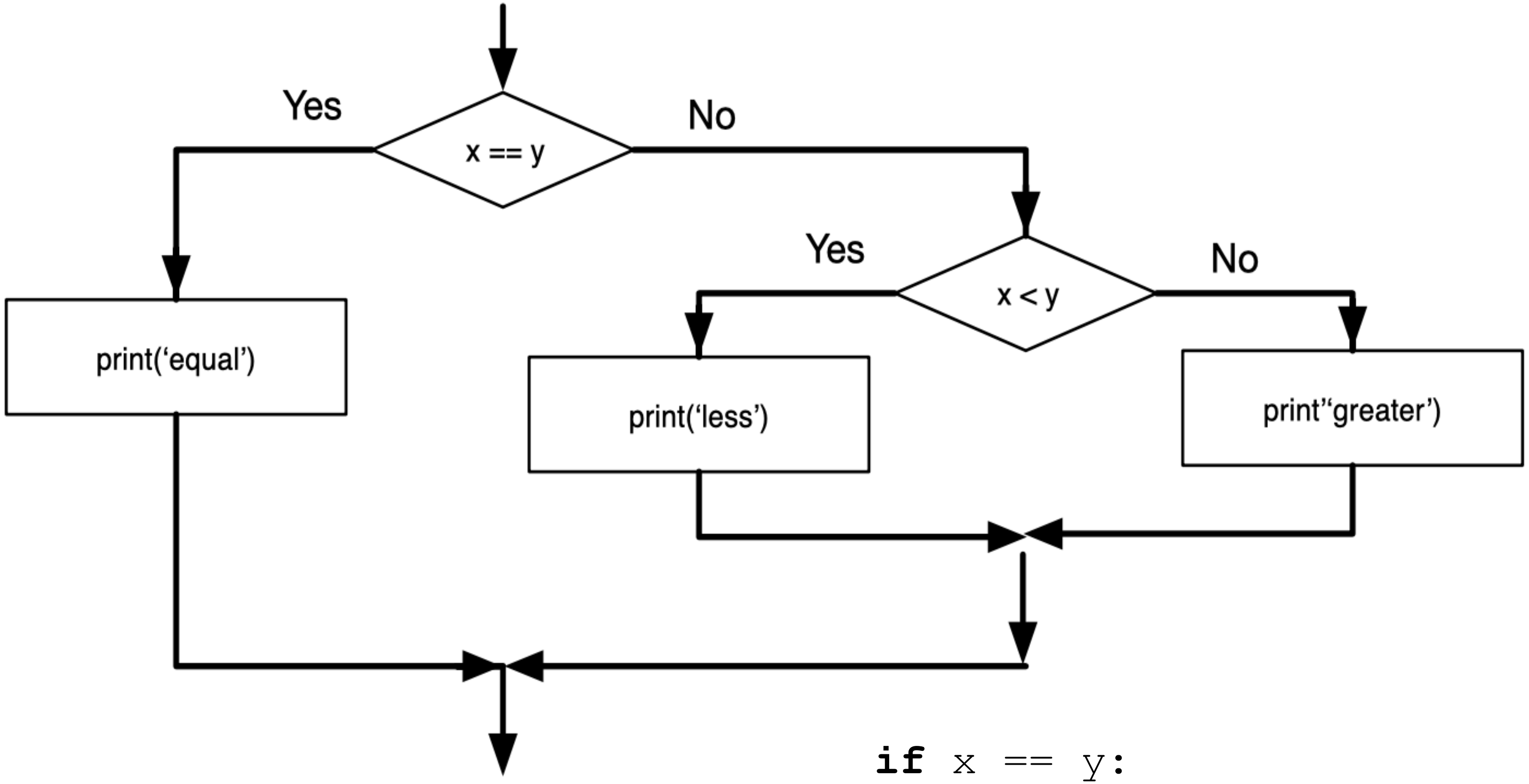
```
if x < 2 :
    print('Below 2')
elif x >= 2 :
    print('Two or more')
else :
    print('Something else')
```

Exercise 3.3
(Objectives 3.5)

Nested Decisions : select another selection statements to run

- One conditional can also be nested with another.

```
if x == y:
    print('x and y are equal’)
else:
    if x < y:
        print('x is less than y’)
    else:
        print('x is greater than y')
```



```
if x == y:
    print('x and y are equal’)
else:
    if x < y:
        print('x is less than y’)
    else:
        print('x is greater than y')
```

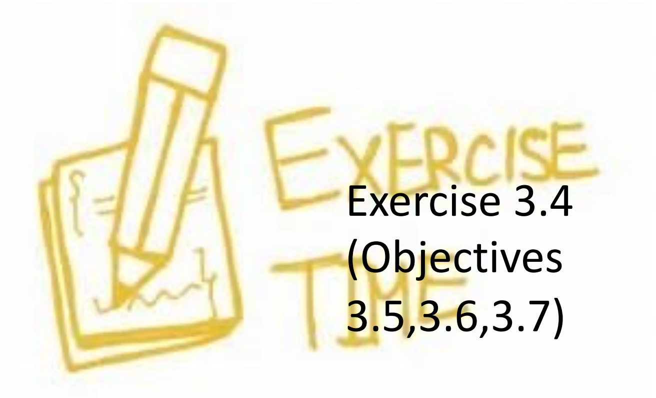
Better to avoid nested decision

```
if   0 < x:
     if x < 10:
          print('x is a positive single-digit number.')
```

After removing nested decision, program becomes easier to read.

```
if 0 < x and x < 10:
     print('x is a positive single-digit number.')
```

How to remove nested decision? -- Logic Operators



Exercise 3.4 (Objectives 3.5,3.6,3.7)

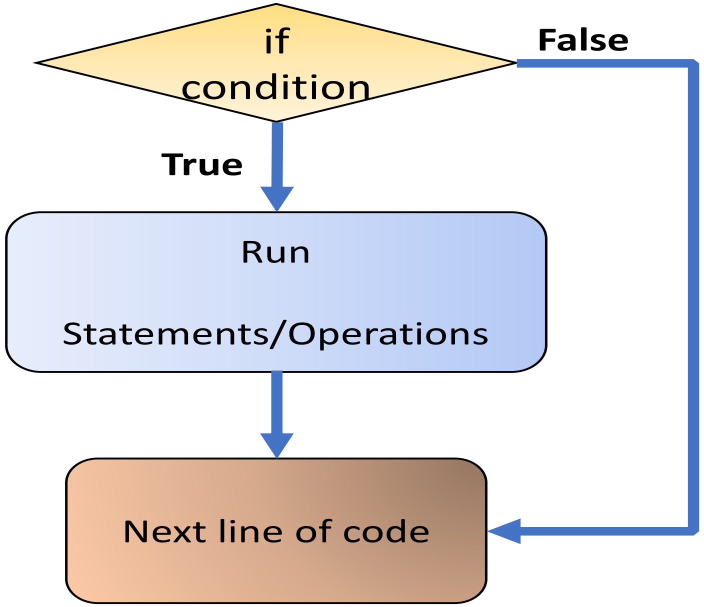
- Three logic operators
- Meaning as the English word
  - x > 0 **and** x < 10

    true only if x is greater than 0 *and* less than 10.
  - n%2 == 0 **or** n%3 == 0

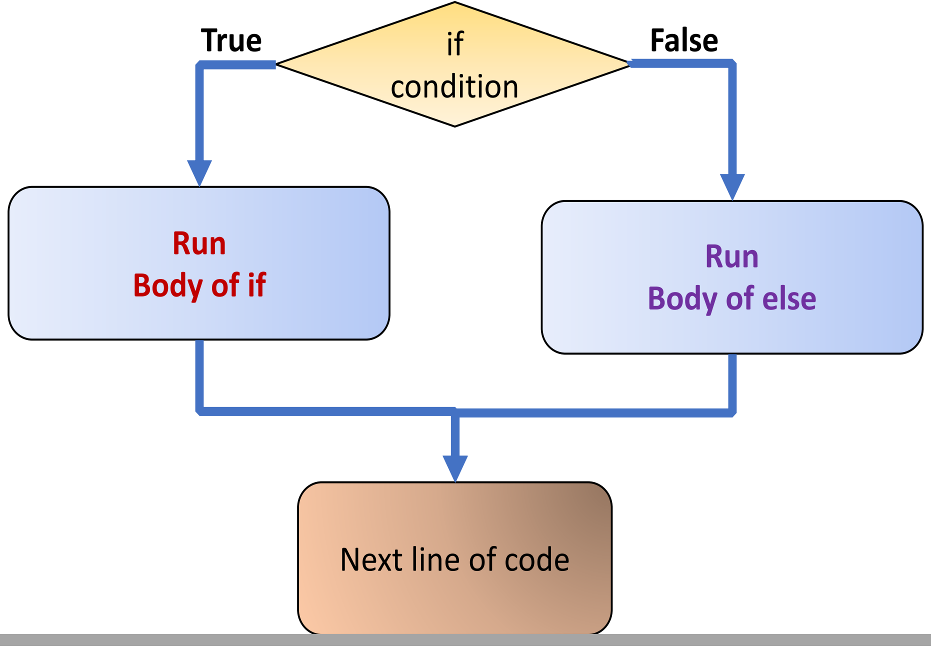
    true if the number is divisible by 2 *or* 3.
  - **not** (x > y)

    true if x is *not* greater than y

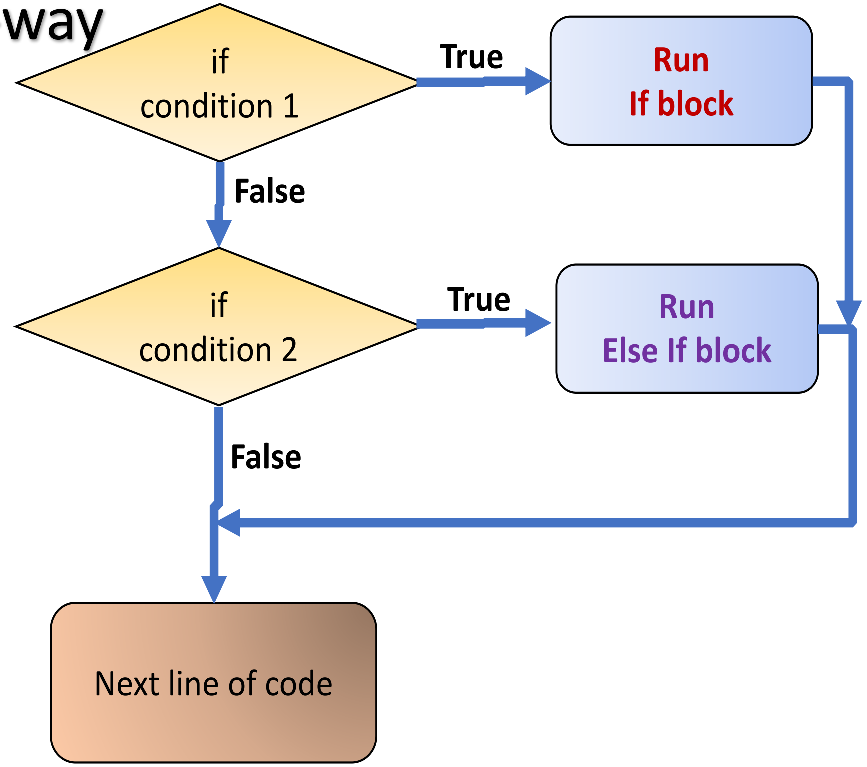
One-way Decision



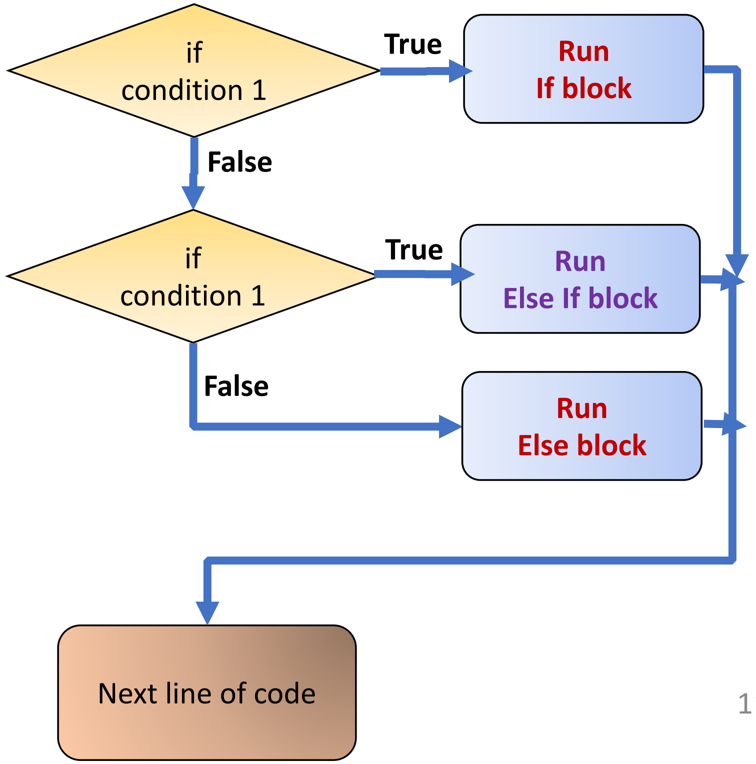
Two-way Decision



Multiple-way Decision (No else)



Multiple-way Decision (With else)



One-way Decision

```
if <condition> :
    <statement>
    <statement>
    <statement>
<Next line of code>
```

Two-way Decision

```
if <condition> :
    <statement>
    <statement>
    <statement>
else :
    <statement>
    <statement>
<Next line of code>
```

Multiple-way Decision (No else)

```
if <condition 1> :
    <statement>
    <statement>
    <statement>
elif <condition 2> :
    <statement>
    <statement>
<Next line of code>
```

Multiple-way Decision (With else)

```
if <condition 1> :
    <statement>
    <statement>
    <statement>
elif <condition 2> :
    <statement>
    <statement>
else:
    <statement>
    <statement>
<Next line of code>
```

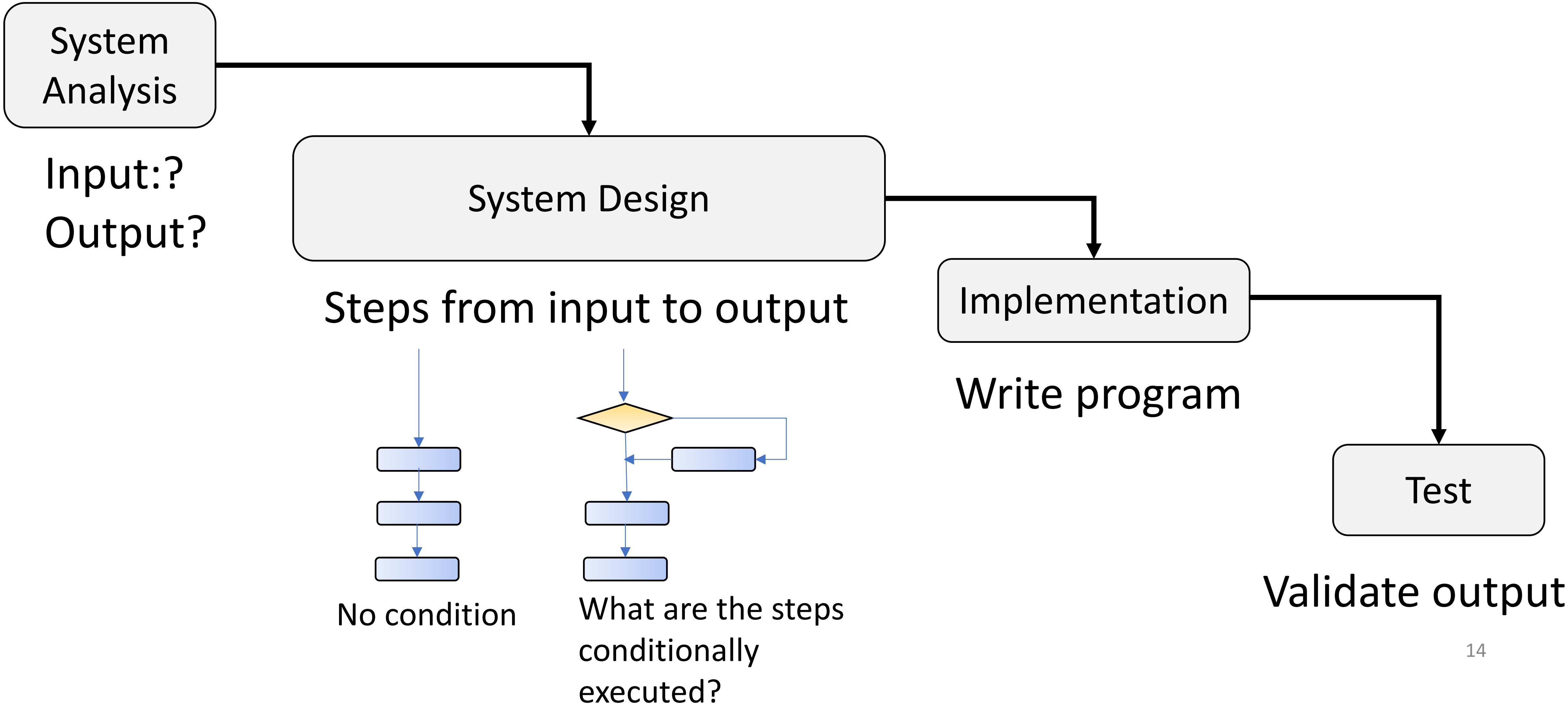
Relational operators for comparison

Python	Meaning
<	Less than
<=	Less than or Equal to
==	Equal to
>=	Greater than or Equal to
>	Greater than
!=	Not equal

Logic operators for avoiding nested selections

Python	Meaning	Example
and	Both LHS and RHS are True?	X>0 and X<100
or	LHS or RHS is True?	X>0 or X<100
not	Opposite of RHS	not (x>0)

Software Development Process



System Analysis

Input:?

Output?

System Design

Steps from input to output

No condition

What are the steps conditionally executed?

Implementation

Write program

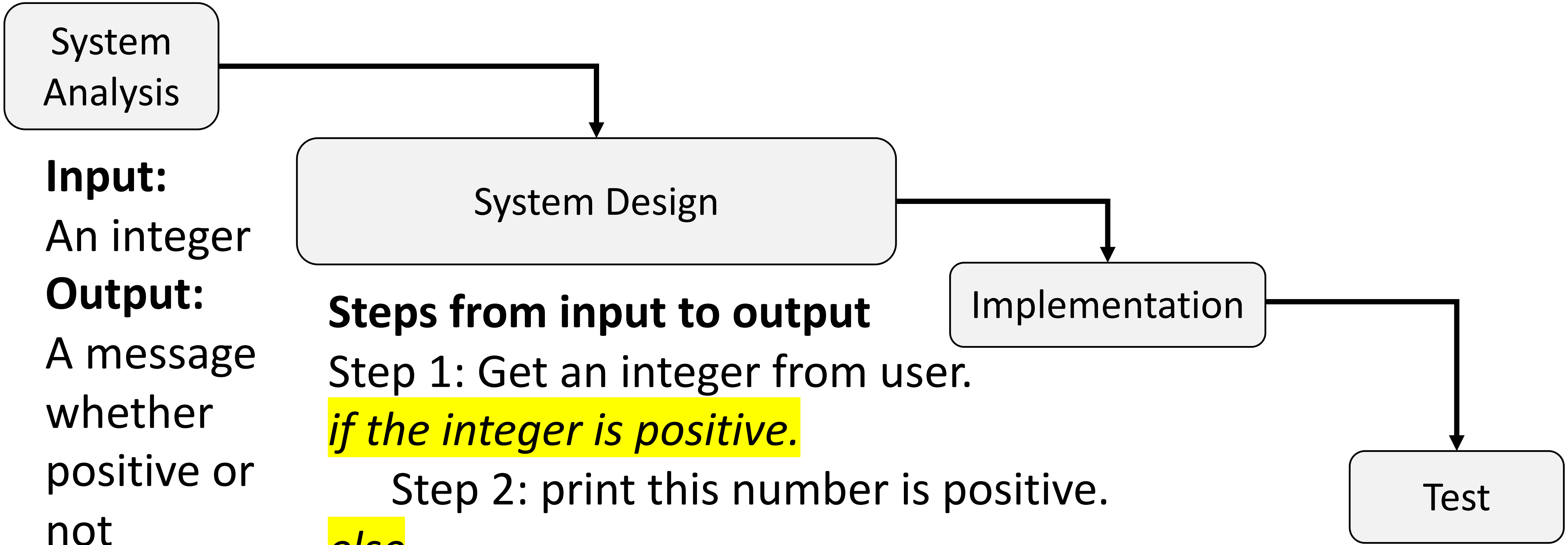
Test

Validate output

Challenge 1 – Check input positive or not

Write a program and ask user to enter an integer, display a message to indicate whether the entered integer is positive or not?

Challenge 1 – System Analysis & Design



Input:

An integer

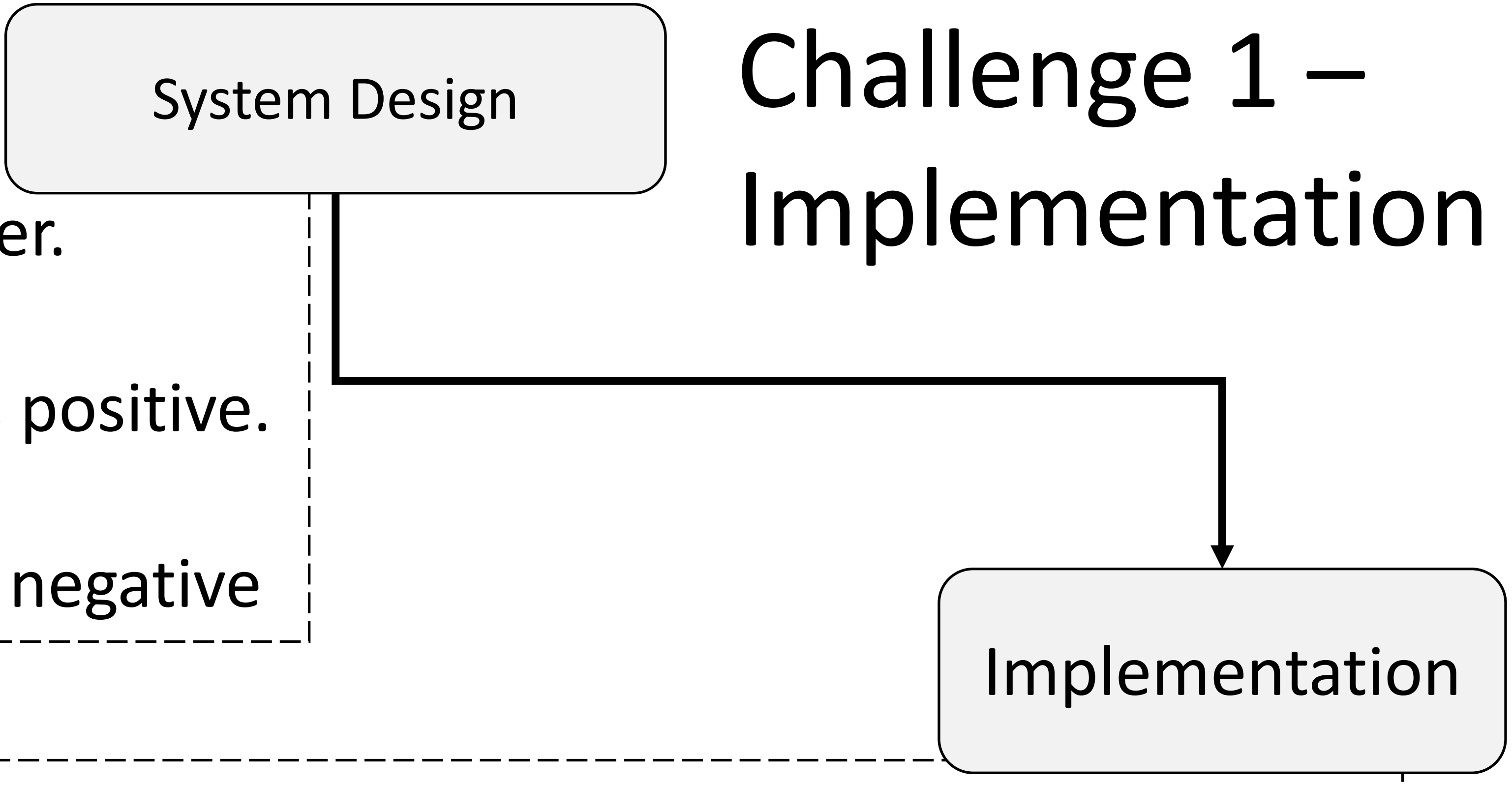
Output:

A message whether positive or not

Steps from input to output

```
Step 1: Get an integer from user.
if the integer is positive.
    Step 2: print this number is positive.
else
    Step 3:print this number is negative
```

Challenge 1 – Implementation

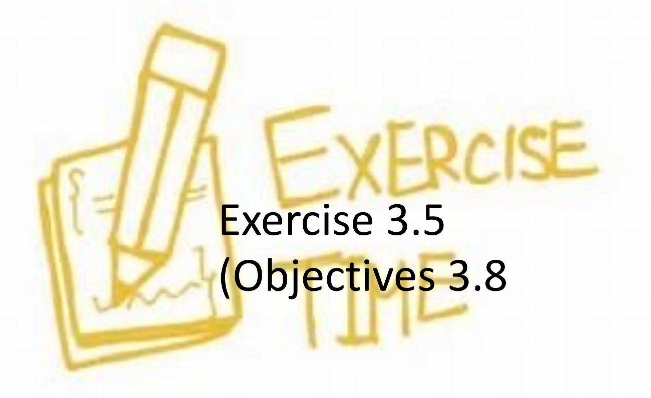


Steps from input to output

```
Step 1: Get an integer from user.
if the integer is positive.
    Step 2: print this number is positive.
else
    Step 3:print this number is negative
```

```
num = int(input("Please enter an integer:"))

if num > 0:
    print(num, "is a positive number")
else:
    print(num, "is a negative number")
```

Challenge 1 - Test



Exercise 3.5
(Objectives 3.8

Test

```
#case 1
Input: 3
Output: 3 is a positive number
```

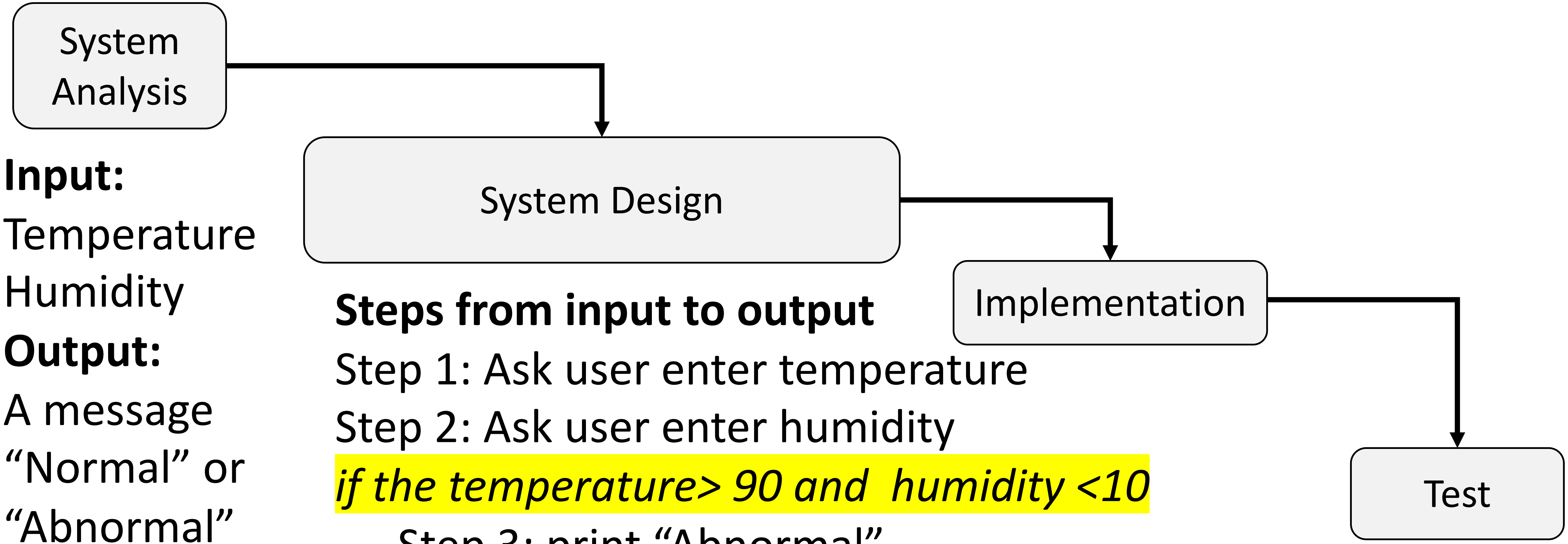
Test

```
#case 2
Input: -3
Output: -3 is a negative number
```

Challenge 2 – Work Environment Check

- Write a program and ask user to enter temperature and humidity. Display a message “Abnormal ” if and only if temperature is greater than 90 and humidity is less than 10; otherwise, display a message “Normal”

Challenge 2 – System Analysis & Design



Input:

Temperature

Humidity

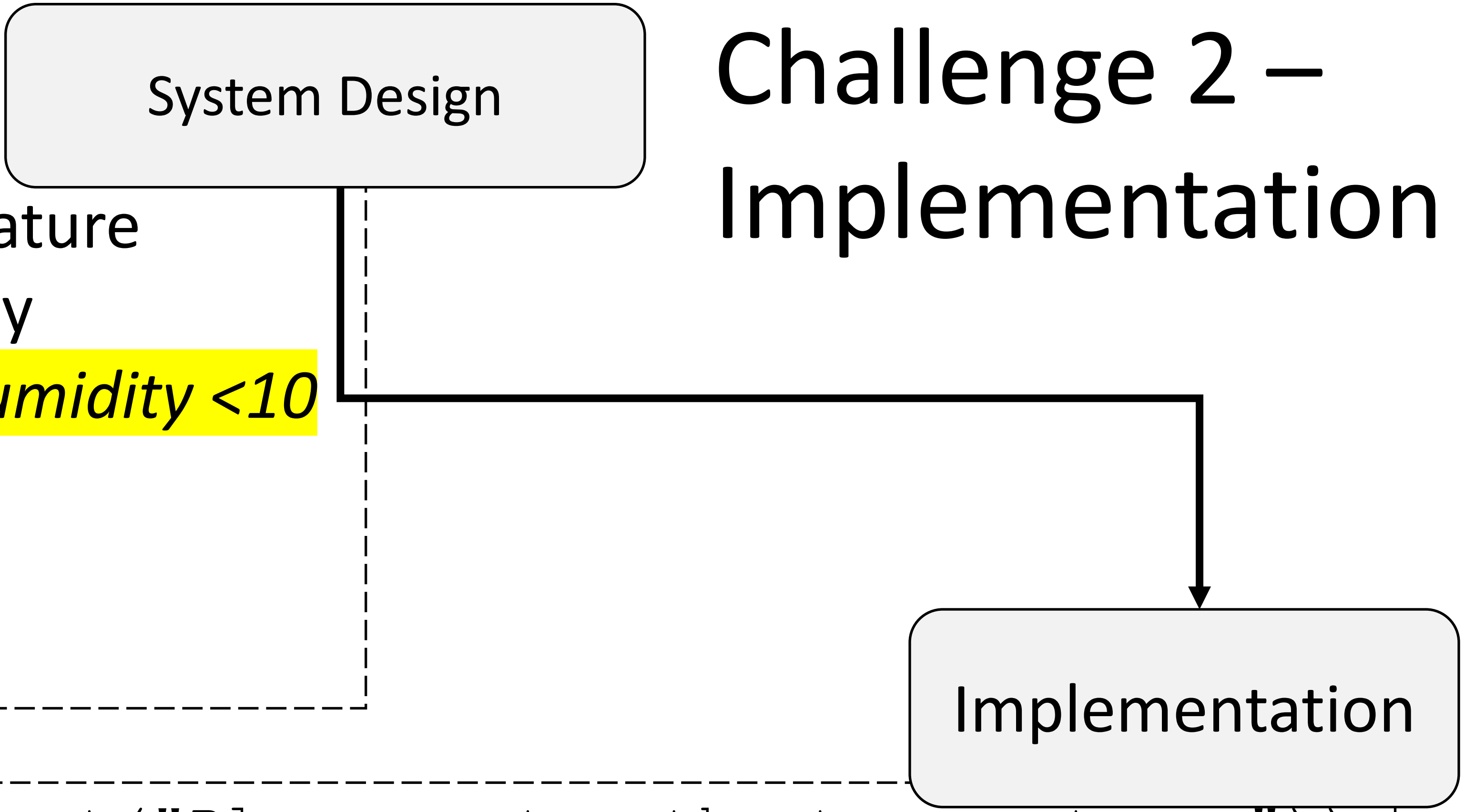
Output:

A message “Normal” or “Abnormal”

Steps from input to output

```
Step 1: Ask user enter temperature
Step 2: Ask user enter humidity
if the temperature> 90 and  humidity <10
    Step 3: print “Abnormal”
else
    Step 4:print “Normal”
```

Challenge 2 – Implementation



Steps from input to output

```
Step 1: Ask user enter temperature
Step 2: Ask user enter humidity
if the temperature> 90 and  humidity <10
    Step 3: print "Abnormal"
else
    Step 4:print "Normal"
```

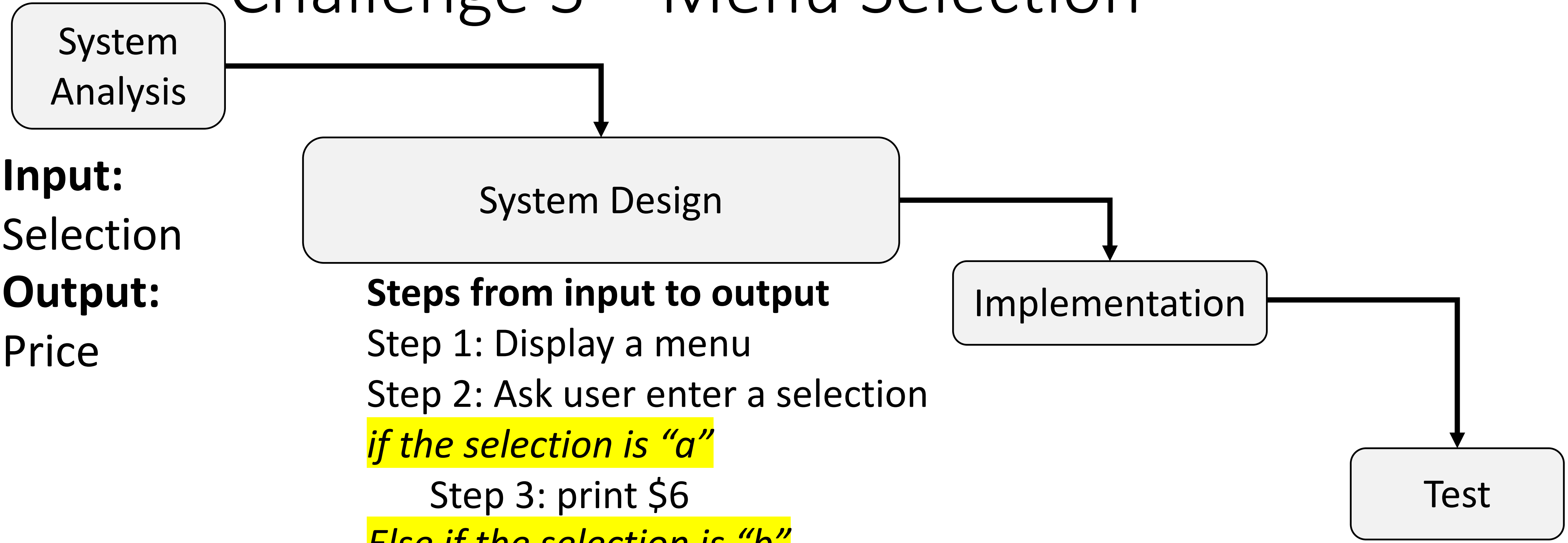
```
temperature = int(input("Please enter the temperature:"))
humidity = int(input("Please enter the humidity:"))

if temperature > 90 and humidity < 10 :
    print("Abnormal")
else:
    print(num, "Normal")
```

Chapter 3 – Menu Selection

- Write a program to display a menu to user and ask user to select a food item to order. Then it prints out the price user needs to pay.

a.Chicken $6
b.Beef $9
c.Shrimp $10
d.Vegetable $5

Challenge 3 – Menu Selection



Input:

Selection

Output:

Price

Steps from input to output

```
Step 1: Display a menu
Step 2: Ask user enter a selection
if the selection is "a"
    Step 3: print $6
Else if the selection is "b"
    Step 4:print $9
Else if the selection is "c"
    Step 4:print $10
Else if the selection is "d"
    Step 4:print $5
```

Challenge 3 – Implementation

Implementation

Exercise 3.6
(Objectives 3.8)

System Design

Steps from input to output

```
Step 1: Display a menu
Step 2: Ask user enter a selection
if the selection is “a”
    Step 3: print $6
Else if the selection is “b”
    Step 4:print $9
Else if the selection is “c”
    Step 4:print $10
Else if the selection is “d”
    Step 4:print $5
```

```python
print("a. Chicken $6")
print("b. Beef $9")
print("c. Shrimp $10")
print("d. Vegetable $5")

select = int(input("Please enter your selection (a-d):"))

if select == 'a' :
    print("Please pay $6")
elif select == 'b' :
    print("Please pay $9")
elif select == 'c' :
    print("Please pay $10")
elif select == 'd' :
    print("Please pay $5")
```

Review of Individual Assignment 1

- How to write an expression correctly in Python?

$$\frac{-10*3+2.5\times 3}{32.6-\frac{13}{9}}$$

```
print((-10 * 3 + 2.5 * 3) / (32.6 - 13/9))
```

$$a = (d1 - d0) / t$$

```
print((d1-d0)/t)
```

Review of Individual Assignment 2

- 1. How to obtain numberOfMonths and montlyInterestRate?

```
numberOfMonths= numberOfYears*12
montlyInterestRate = (annualInterestRate /100) /12
```

- 2. How to calculate gratuity and total?

```
gratuity = subtotal*(rate/100)
total = subtotal+ gratuity
```

Review of Quiz 1

- Write a single-line **print** statement to display the result of πr^4 when $r=2.5$, $\pi=3.14$

```
print(3.14*2.5*2.5*2.5*2.5)

print(3.14*(2.5**4))

print(3.14*2.5^4)  # wrong
```

Objectives covered in Midterm

---Chapter 1--

1.1 To explain and describe the concepts of computer hardware, programs, and operating systems (§1.2 -1.4)

1.2 To describe the history of Python (§1.5)

1.3 To explain the basic syntax of a Python program (§1.6)

1.4 To write and run a simple Python program (§1.6)

1.5 To use sound programming style and document programs properly (§1.7).

1.6 To explain the differences between syntax errors, runtime errors, and logic errors (§1.8).

--- Chapter 2---

2.1 To write programs that perform simple computations (§2.2)

2.2 To obtain input from a program’s user by using the input function and to convert strings to numbers using the int and float functions (§2.3)

2.3 To use identifiers to name elements such as variables and functions (§2.4)

2.4 To assign data to variables (§2.5)

2.5 To define named constants (§2.7)

2.6 To use the operators +, −, *, /, //, %, and ** (§2.8)

2.7 To program using division and remainder operators (§2.9)

2.8 To write and evaluate numeric expressions (§2.10)

2.9 To use augmented assignment operators to simplify coding (§2.11)

2.10 To perform numeric type conversion and rounding with the round function (§2.12)

2.11 To describe the software development process and apply it to develop the loan payment program (§2.14)

--- Chapter 3 ---

3.1 To write Boolean expressions using relational operators (§3.2).

3.2 To program with Boolean expressions (§3.3).

3.3 To implement selection control using one-way if statements (§3.4).

3.4 To implement selection control using two-way if-else statements (§3.5).

3.5 To implement selection control with nested if and multi-way if-elif-else statements (§3.6).

3.6 To combine conditions using logical operators (and, or, and not) (§3.10).

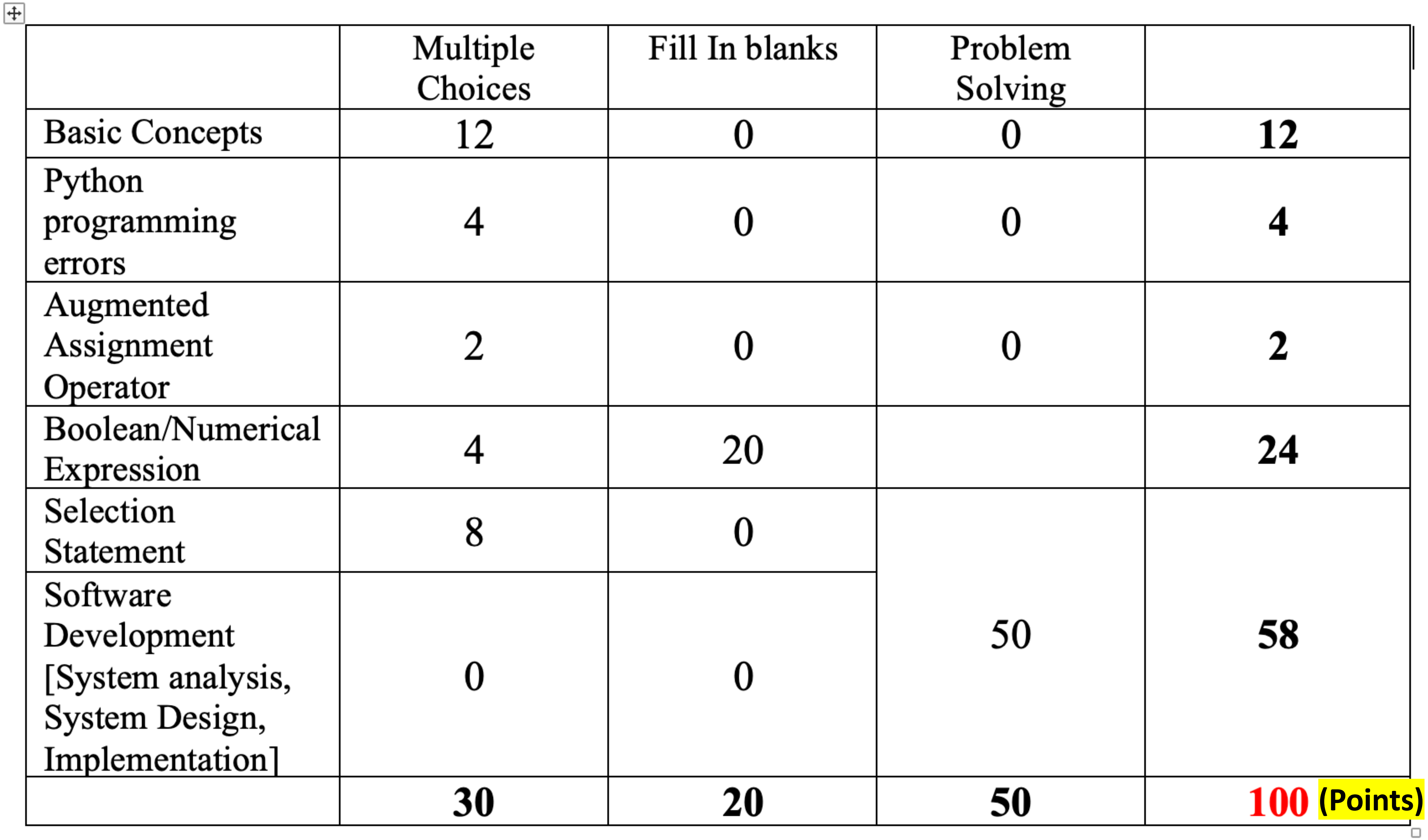
3.7 To use selection statements with combined conditions (§§3.11–3.12).

3.8 To understand how to develop a program with selections.

Midterm : Style of Questions

- Multiple Choices
- Fill in blanks
- Software development
    - System Analysis
    - System Design
    - Implementation

Test Design Matrix



	Multiple Choices	Fill In blanks	Problem Solving	
Basic Concepts	12	0	0	**12**
Python programming errors	4	0	0	**4**
Augmented Assignment Operator	2	0	0	**2**
Boolean/Numerical Expression	4	20		**24**
Selection Statement	8	0	50	**58**
Software Development [System analysis, System Design, Implementation]	0	0		
	30	**20**	**50**	**100** **(Points)**

Midterm : How to Prepare?

- Review slides from week 1 to week 5
- Review in-class exercises
- Review individual assignments 1 to 3
- Review quiz 1 and quiz 2