Objectives for class 4

- --- Chapter 2 ---

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).

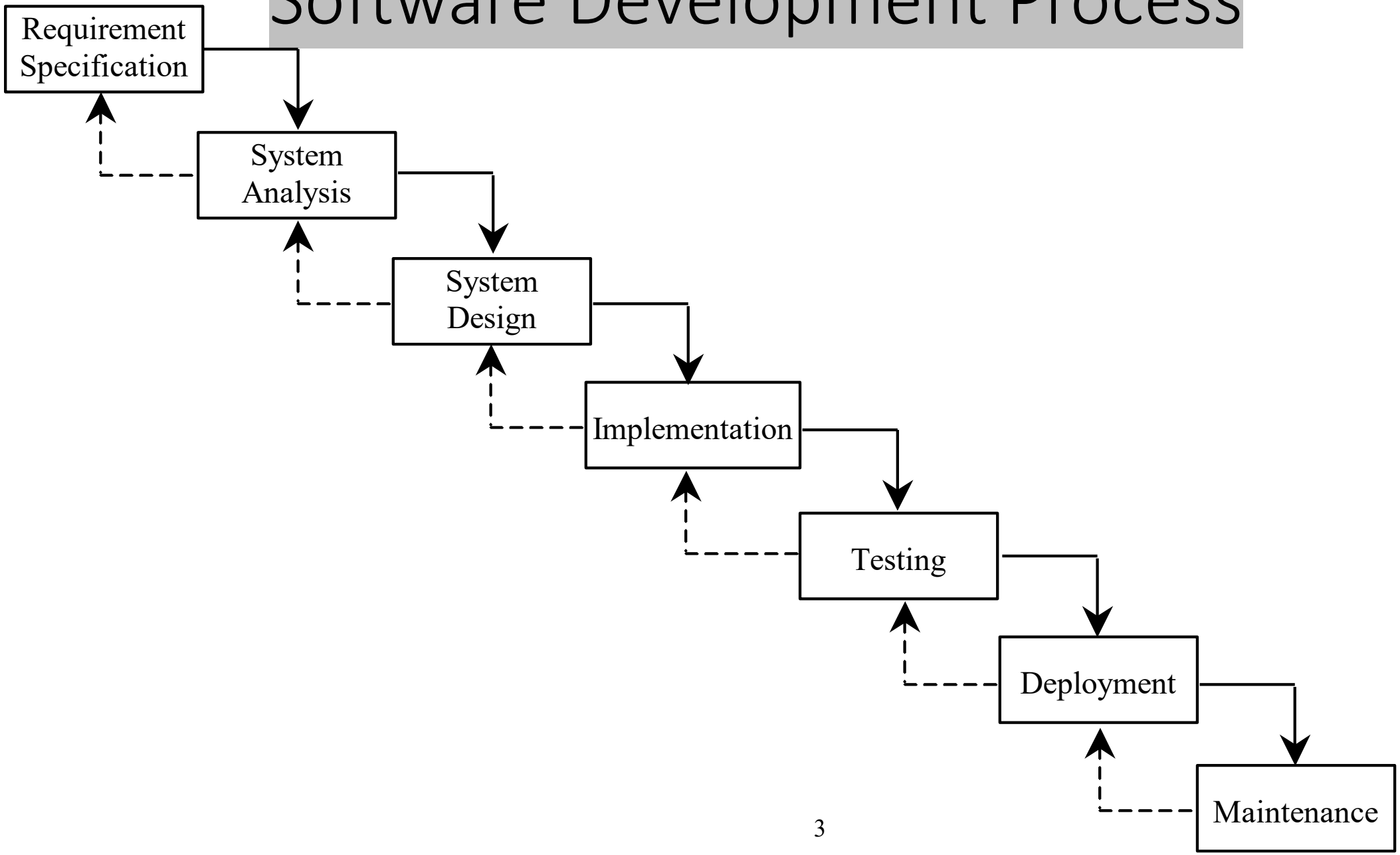
3.5 To implement selection control with nested if and multi-way if-elif-elsestatements (§3.6).

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).

- Statements
  - print(x)
  - X=Input()
  - X=x+1
  - x
- Function [what does it do? What does it return? What does it need?]
  - A=Float(“123.3”)
  - Int(“12”)+1
  - Float(Input())
  - Print()
  - Str()
  - X()

Software Development Process



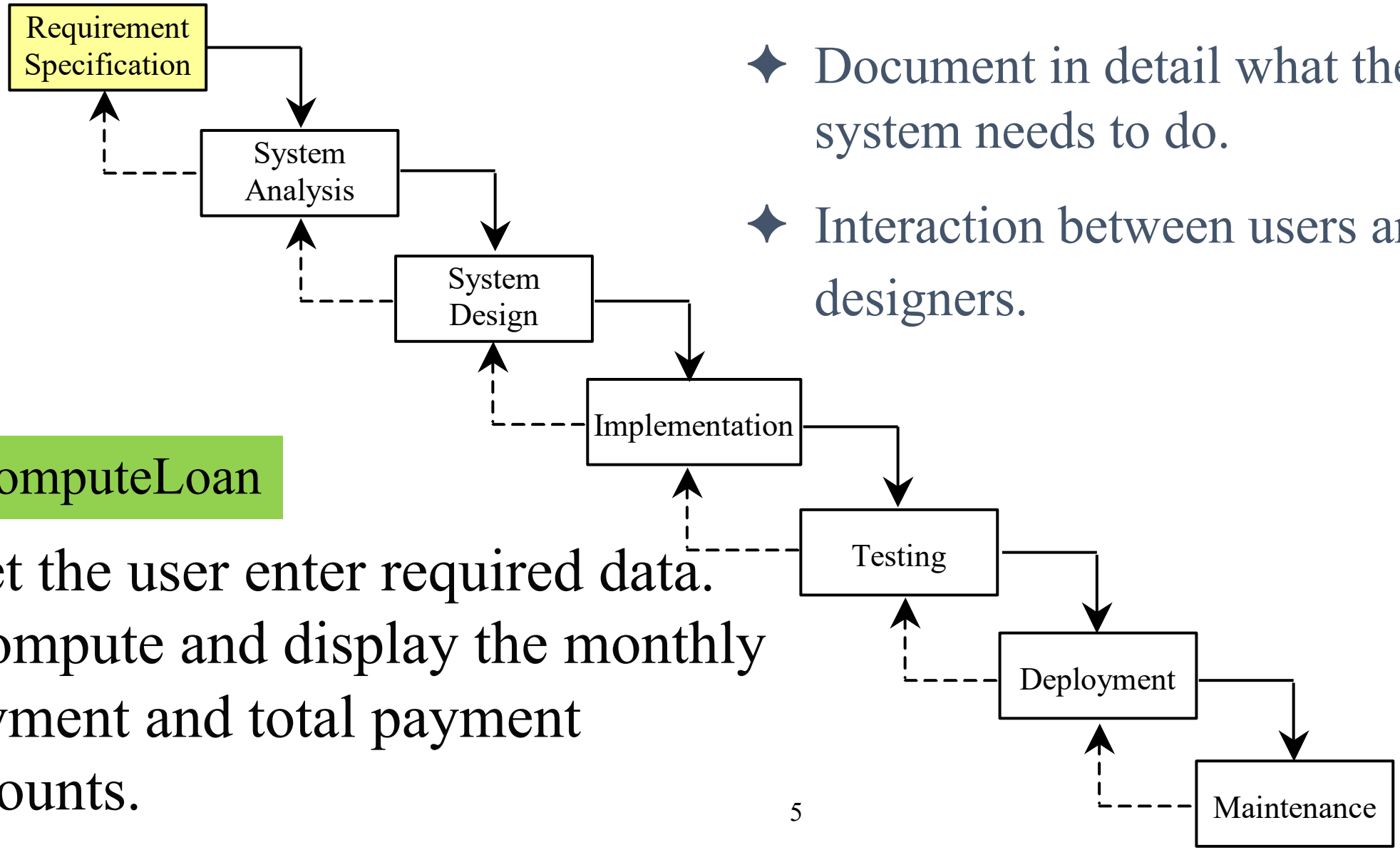
Problem: Computing Loan Payments

This program lets the user enter the interest rate, number of years, and loan amount, and computes monthly payment and total payment.

$$monthlyPayment = \frac{loanAmount \times monthlyInterestRate}{1 - \frac{1}{(1 + monthlyInterestRate)^{numberOfYears \times 12}}}$$

ComputeLoan

Requirement Specification : Document what the software needs to do

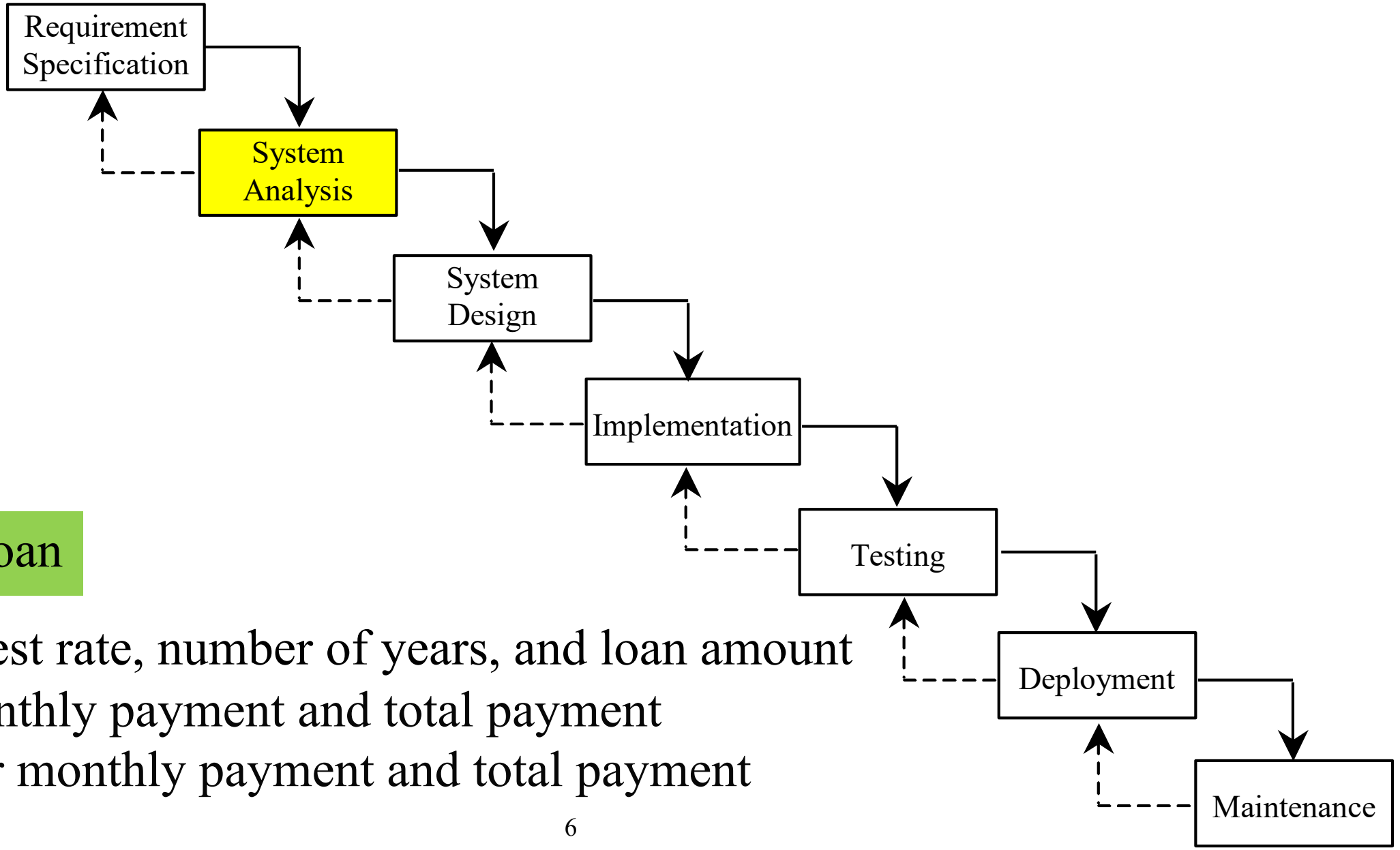


- Document in detail what the software system needs to do.
- Interaction between users and designers.

ComputeLoan

- Let the user enter required data.
- Compute and display the monthly payment and total payment amounts.

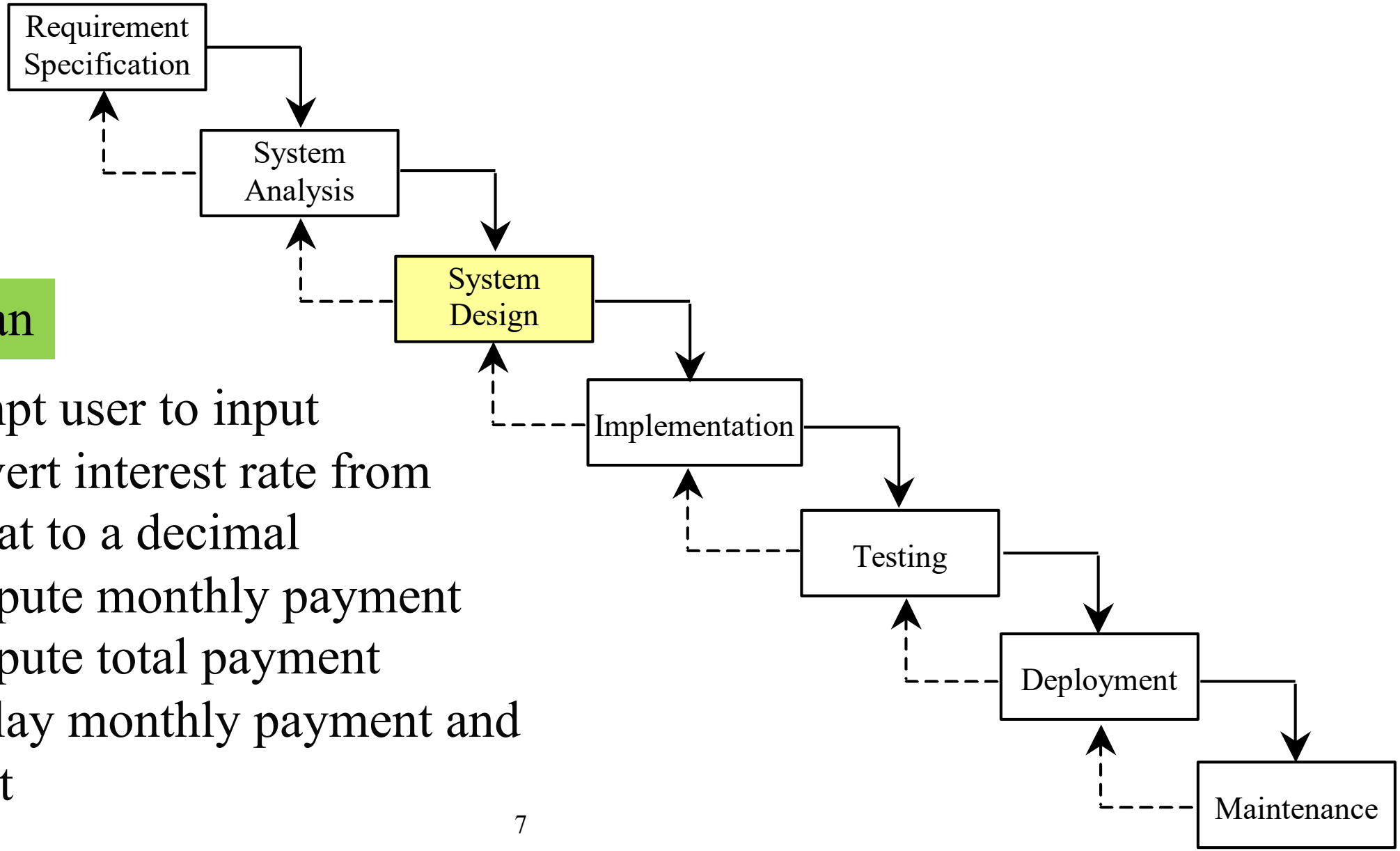
System Analysis : Define input and output



ComputeLoan

- Input: interest rate, number of years, and loan amount
- Output: monthly payment and total payment
- Formula for monthly payment and total payment

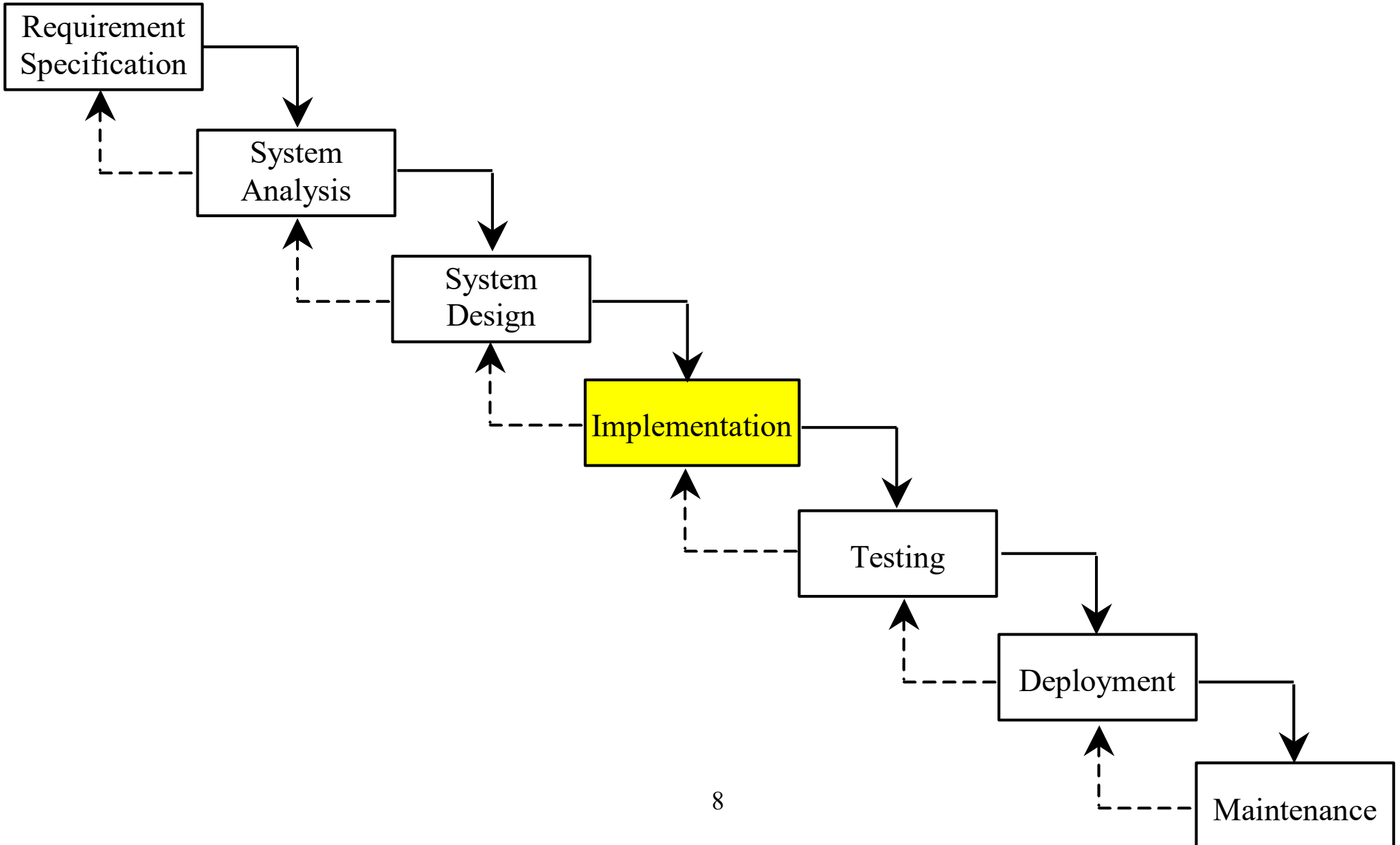
System Design : Steps from Input to Output



ComputeLoan

- Step1: Prompt user to input
- Step2: Convert interest rate from percent format to a decimal
- Step3: Compute monthly payment
- Step4: Compute total payment
- Step5: Display monthly payment and total payment

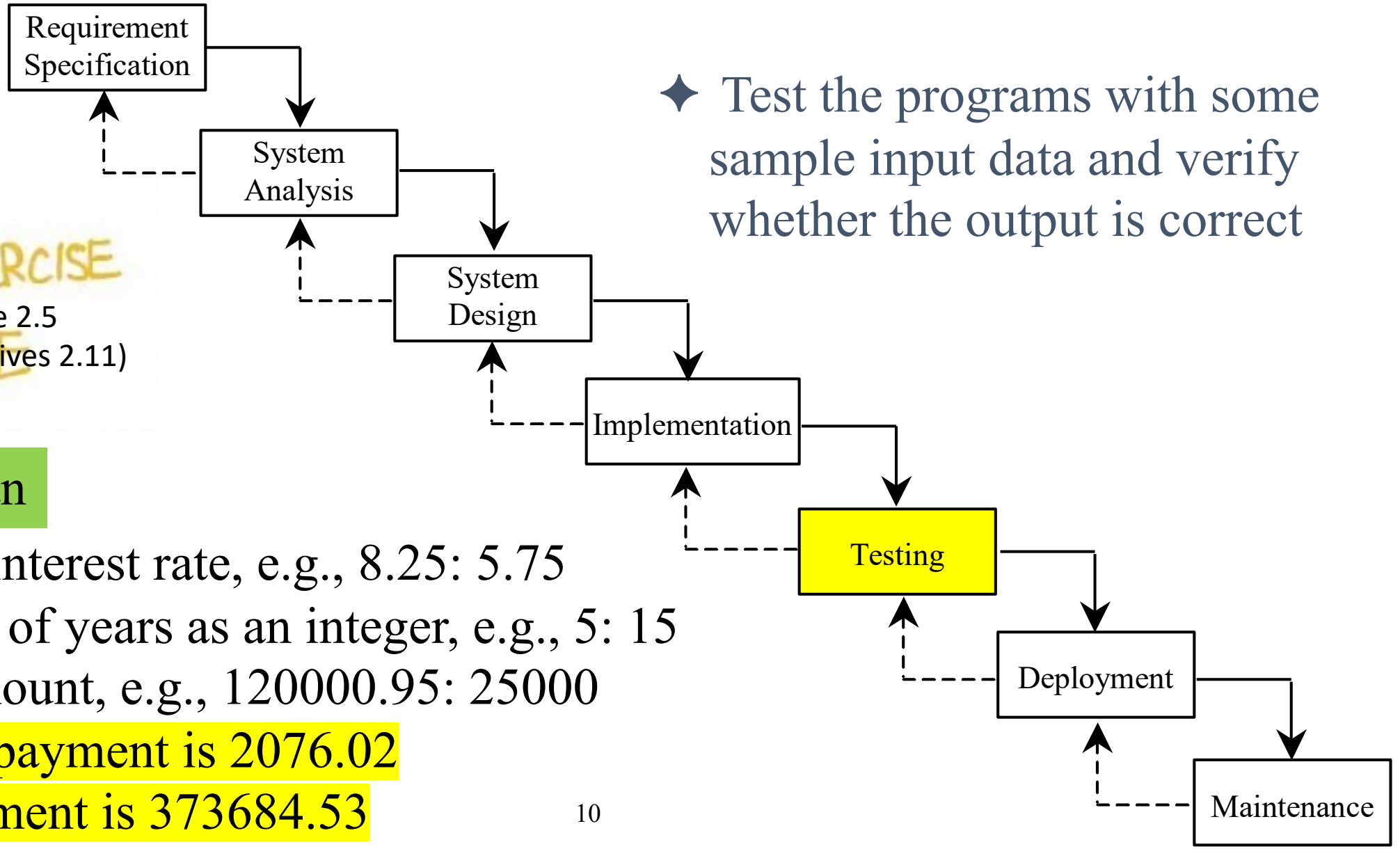
Implementation : Write Programs



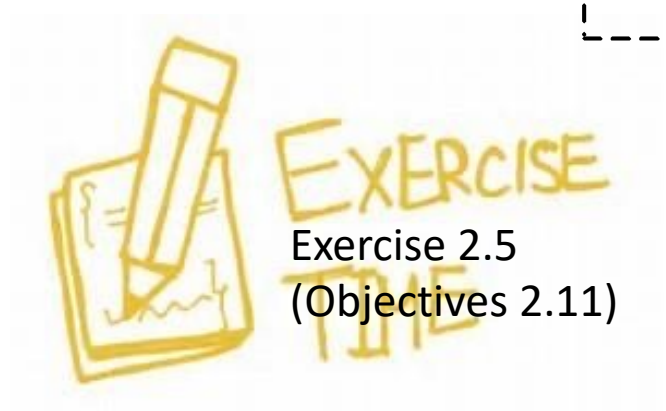
ComputeLoan

```
1  # Enter annual interest rate
2  annualInterestRate = float(input(
3    "Enter annual interest rate, e.g., 8.25: "))
4  monthlyInterestRate = annualInterestRate / 1200
5
6  # Enter number of years
7  numberOfYears = int(input(
8    "Enter number of years as an integer, e.g., 5: "))
9
10 # Enter loan amount
11 loanAmount = float(input("Enter loan amount, e.g., 120000.95: "))
12
13 # Calculate payment
14 monthlyPayment = loanAmount * monthlyInterestRate / (1
15   - 1 / (1 + monthlyInterestRate) ** (numberOfYears * 12))
16 totalPayment = monthlyPayment * numberOfYears * 12
17
18 # Display results
19 print("The monthly payment is", int(monthlyPayment * 100) / 100)
20 print("The total payment is", int(totalPayment * 100) /100)
```

Testing : Verify if Output Correct



- Test the programs with some sample input data and verify whether the output is correct



Exercise 2.5
(Objectives 2.11)

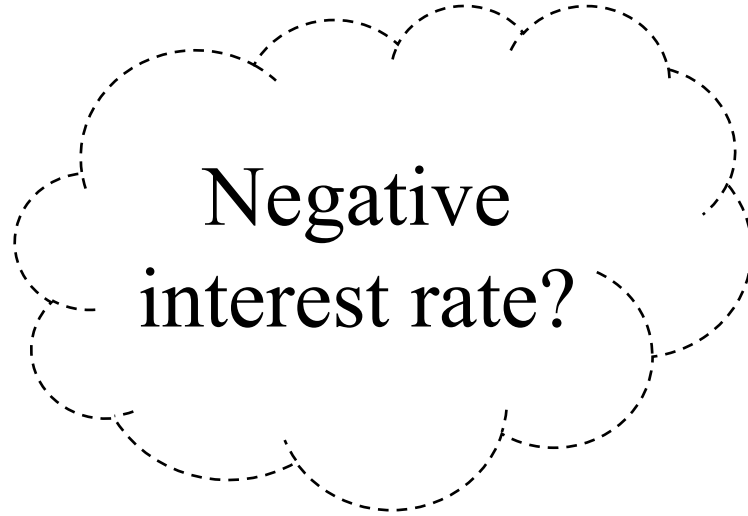
ComputeLoan

Enter annual interest rate, e.g., 8.25: 5.75
Enter number of years as an integer, e.g., 5: 15
Enter loan amount, e.g., 120000.95: 25000
The monthly payment is 2076.02
The total payment is 373684.53

Selections in a program - Select Statement(s) to Run

ComputeLoan

- Step1: Prompt user to input
- Step2: Convert interest rate from percent format to a decimal
- Step3: Compute monthly payment
- Step4: Compute total payment
- Step5: Display monthly payment and total payment



Selections in a program - Select one block to Run

ComputeLoan

- Step1: Prompt user to input

Only when interest rate >= 0, run step2 to step 5

- Step2: Convert interest rate from percent format to a decimal
- Step3: Compute monthly payment
- Step4: Compute total payment
- Step5: Display monthly payment and total payment

Selections in a program - Select either block to Run

ComputeLoan

•Step1: Prompt user to input

If interest rate >= 0, run step2 to step 5

- Step2: Convert interest rate from percent format to a decimal
- Step3: Compute monthly payment
- Step4: Compute total payment
- Step5: Display monthly payment and total payment

Otherwise, run step 6

- Step 6: Display error message

Negative interest rate?

Execution Flow

```
1  # Enter annual interest rate
2  annualInterestRate = float(input(
3    "Enter annual interest rate, e.g., 8.25: "))
4  monthlyInterestRate = annualInterestRate / 1200
5
6  # Enter number of years
7  numberOfYears = int(input(
8    "Enter number of years as an integer, e.g., 5: "))
9
10 # Enter loan amount
11 loanAmount = float(input("Enter loan amount, e.g., 120000.95: "))
12
13 # Calculate payment
14 monthlyPayment = loanAmount * monthlyInterestRate / (1
15   - 1 / (1 + monthlyInterestRate) ** (numberOfYears * 12))
16 totalPayment = monthlyPayment * numberOfYears * 12
17
18 # Display results
19 print("The monthly payment is", int(monthlyPayment * 100) / 100)
20 print("The total payment is", int(totalPayment * 100) /100)
```

No Selection

Execution Flow

Has Selection

```
# Enter yearly interest rate
annualInterestRate = float(input(
   "Enter annual interest rate, e.g., 8.25: "))
monthlyInterestRate = annualInterestRate / 1200

# Enter number of years
numberOfYears = int(input(
   "Enter number of years as an integer, e.g., 5: "))
# Enter loan amount
loanAmount = float(input("Enter loan amount, e.g., 120000.95: "))

# Calculate payment
if annualInterestRate>=0:
    monthlyPayment = loanAmount * monthlyInterestRate / (1
       - 1 / (1 + monthlyInterestRate) ** (numberOfYears * 12))
    totalPayment = monthlyPayment * numberOfYears * 12

    # Display results
    print("The monthly payment is", int(monthlyPayment * 100) / 100)
    print("The total payment is", int(totalPayment * 100) /100)
```

Execution Flow

Has Selection

```
# Enter yearly interest rate
annualInterestRate = float(input(
   "Enter annual interest rate, e.g., 8.25: "))
monthlyInterestRate = annualInterestRate / 1200

# Enter number of years
numberOfYears = int(input(
   "Enter number of years as an integer, e.g., 5: "))
# Enter loan amount
loanAmount = float(input("Enter loan amount, e.g., 120000.95: "))

# Calculate payment
if annualInterestRate>=0:
    monthlyPayment = loanAmount * monthlyInterestRate / (1
       - 1 / (1 + monthlyInterestRate) ** (numberOfYears * 12))
    totalPayment = monthlyPayment * numberOfYears * 12

    # Display results
    print("The monthly payment is", int(monthlyPayment * 100) / 100)
    print("The total payment is", int(totalPayment * 100) /100)
else:
    print("Please enter a positive interest rate")
```

Boolean Expressions – Represent a condition

- **Boolean expressions** ask a question and produce a YES or NO result

- Boolean expressions using **relational operators**

Boolean Type – Represents YES or NO

- ***Boolean*** data type has only two values: True and False
- No quotes around *True* and *False*
- Start with a capital **T** or **F**, with the rest of the word in lowercase

```
>>> spam = True
>>> spam
True
>>> true
Traceback (most recent call
last):
 File "<pyshell#2>", line 1,
in <module>
   true
NameError: name 'true' is not
defined
>>> True = 2 + 2
SyntaxError: can't assign to
keyword
```

Relational Operators for Comparison

- Relational operators look at variables but do not change the variables

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

Remember: "=" is used for assignment.

Relational Operator Examples

```
>>> 42 == 42
True
>>> 42 == 99
False
>>> 2 != 3
True
>>> 2 != 2
False
```

```
>>> 42 < 100
True
>>> 42 > 100
False
>>> 42 < 42
False
>>> eggCount = 42
>>> eggCount <= 42
True
>>> myAge = 29
>>> myAge >= 10
True
```

```
>>> 'hello' == 'hello'
True
>>> 'hello' == 'Hello'
False
>>> 'dog' != 'cat'
True
>>> True == True
True
>>> True != False
True
>>> 42 == 42.0
True
>>> 42 == '42'
False
```

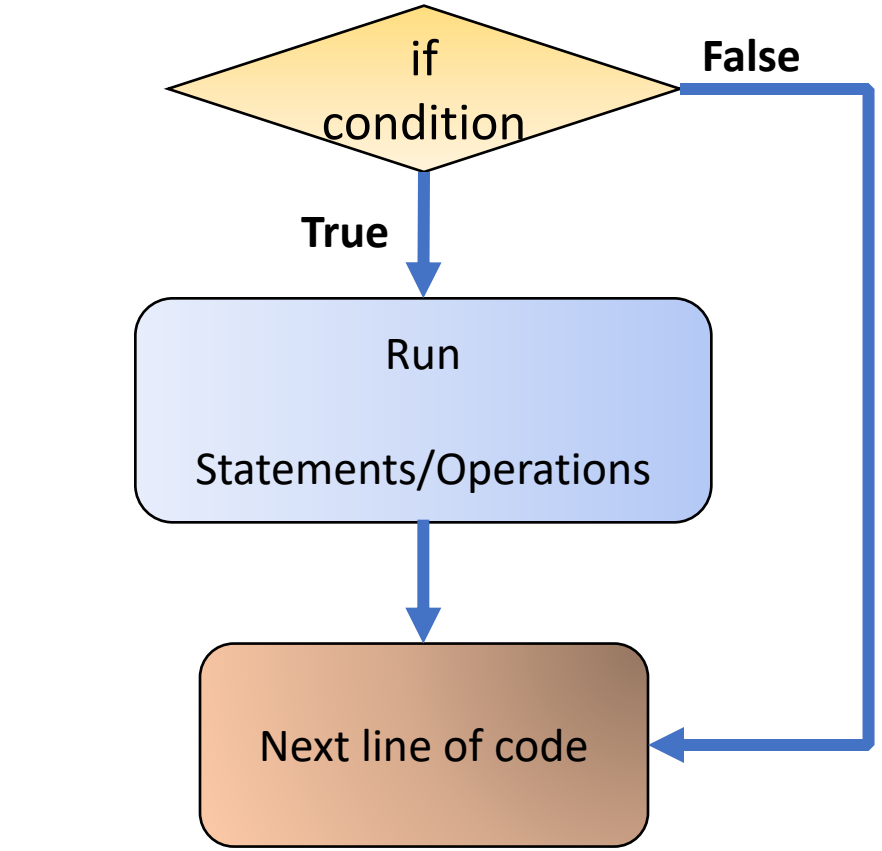
Exercise 3.1
(Objectives 3.1)

Conditional Execution

- One-Way Decisions
- Two-Way Decisions
- Multiple-Way Decisions

One-Way Decisions : select one block to execute

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



Execution Flow

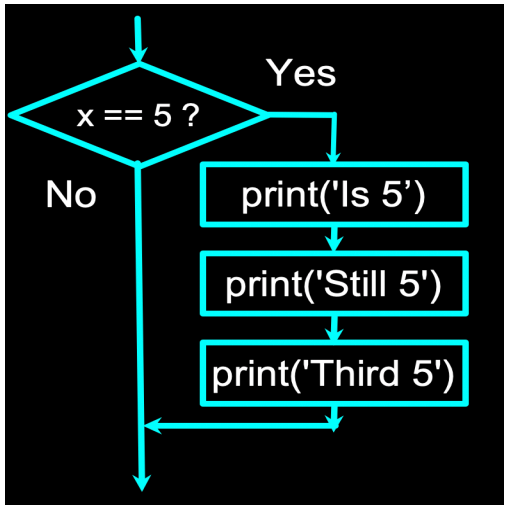
One-way Decision

```
# Enter yearly interest rate
annualInterestRate = float(input(
    "Enter annual interest rate, e.g., 8.25: "))
monthlyInterestRate = annualInterestRate / 1200

# Enter number of years
numberOfYears = int(input(
    "Enter number of years as an integer, e.g., 5: "))
# Enter loan amount
loanAmount = float(input("Enter loan amount, e.g., 120000.95: "))

# Calculate payment
if annualInterestRate>=0:
    monthlyPayment = loanAmount * monthlyInterestRate / (1
        - 1 / (1 + monthlyInterestRate) ** (numberOfYears * 12))
    totalPayment = monthlyPayment * numberOfYears * 12

    # Display results
    print("The monthly payment is", int(monthlyPayment * 100) / 100)
    print("The total payment is", int(totalPayment * 100) /100)
```

True! RUN ->

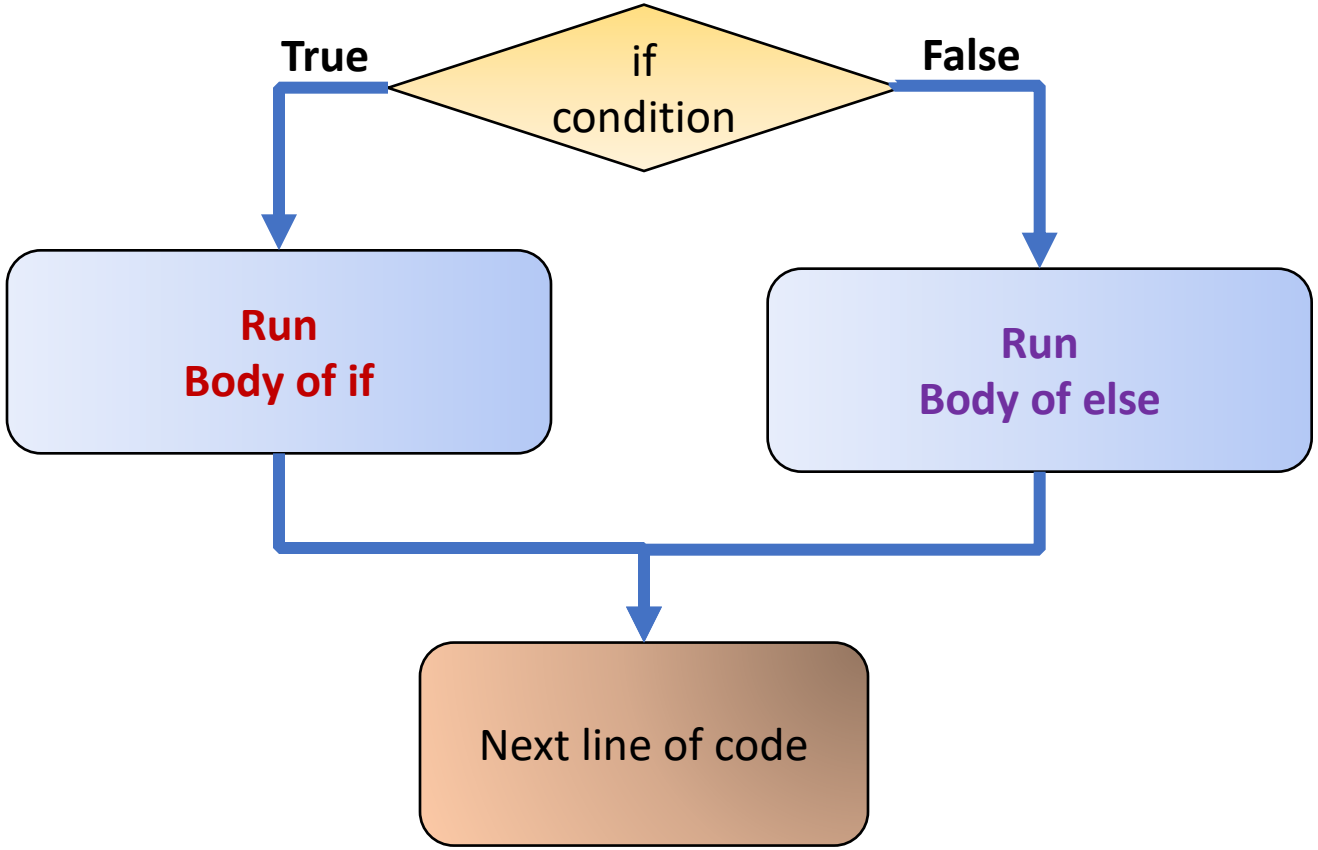
```
1  x = 5
2  print('Before 5')
3  if  x == 5 :
4      print('Is 5')
5      print('Is Still 5')
6      print('Third 5')
7  print('Afterwards 5')
8  print('Before 6')
9  if x == 6 :
10     print('Is 6')
11     print('Is Still 6')
12     print('Third 6')
13 print('Afterwards 6')
```

```
Before 5
Is 5
Is Still 5
Third 5
Afterwards 5
Before 6
Afterwards 6
```



Two-way Decisions : select either block to execute

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



Execution Flow

Has Selection

```
# Enter yearly interest rate
annualInterestRate = float(input(
   "Enter annual interest rate, e.g., 8.25: "))
monthlyInterestRate = annualInterestRate / 1200

# Enter number of years
numberOfYears = int(input(
   "Enter number of years as an integer, e.g., 5: "))
# Enter loan amount
loanAmount = float(input("Enter loan amount, e.g., 120000.95: "))

# Calculate payment
if annualInterestRate>=0:
    monthlyPayment = loanAmount * monthlyInterestRate / (1
       - 1 / (1 + monthlyInterestRate) ** (numberOfYears * 12))
    totalPayment = monthlyPayment * numberOfYears * 12

    # Display results
    print("The monthly payment is", int(monthlyPayment * 100) / 100)
    print("The total payment is", int(totalPayment * 100) /100)
else:
    print("Please enter a positive interest rate")
```

True! RUN ->

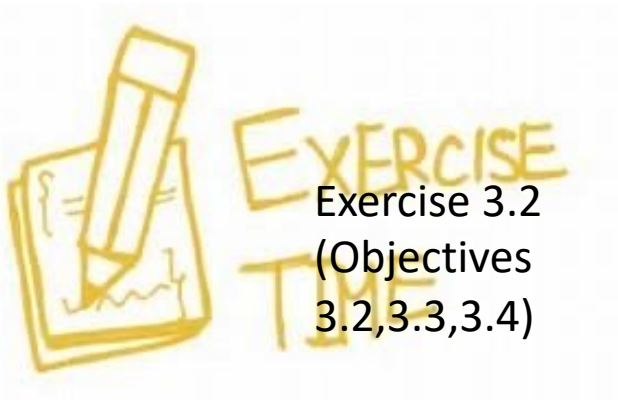
False! RUN ->

Practice

- What is the output of the code in a) and b) if number is 30?

```
(a)
if number % 2 == 0:
    print(number, "is even.")

print(number, "is odd.")
```

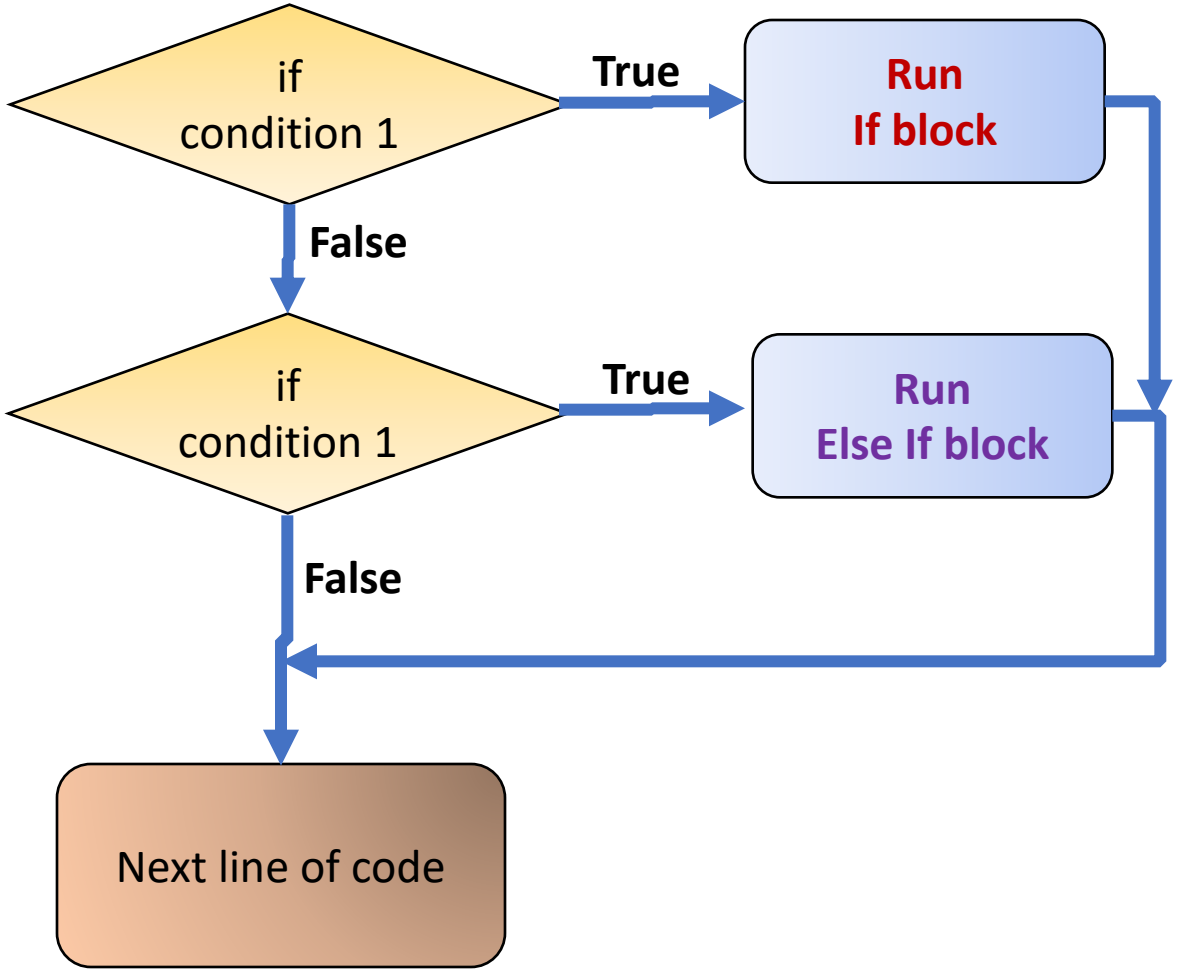
```
(b)
if number % 2 == 0:
    print(number, "is even.")
else:
    print(number, "is odd.")
```



Exercise 3.2 (Objectives 3.2,3.3,3.4)

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>
```

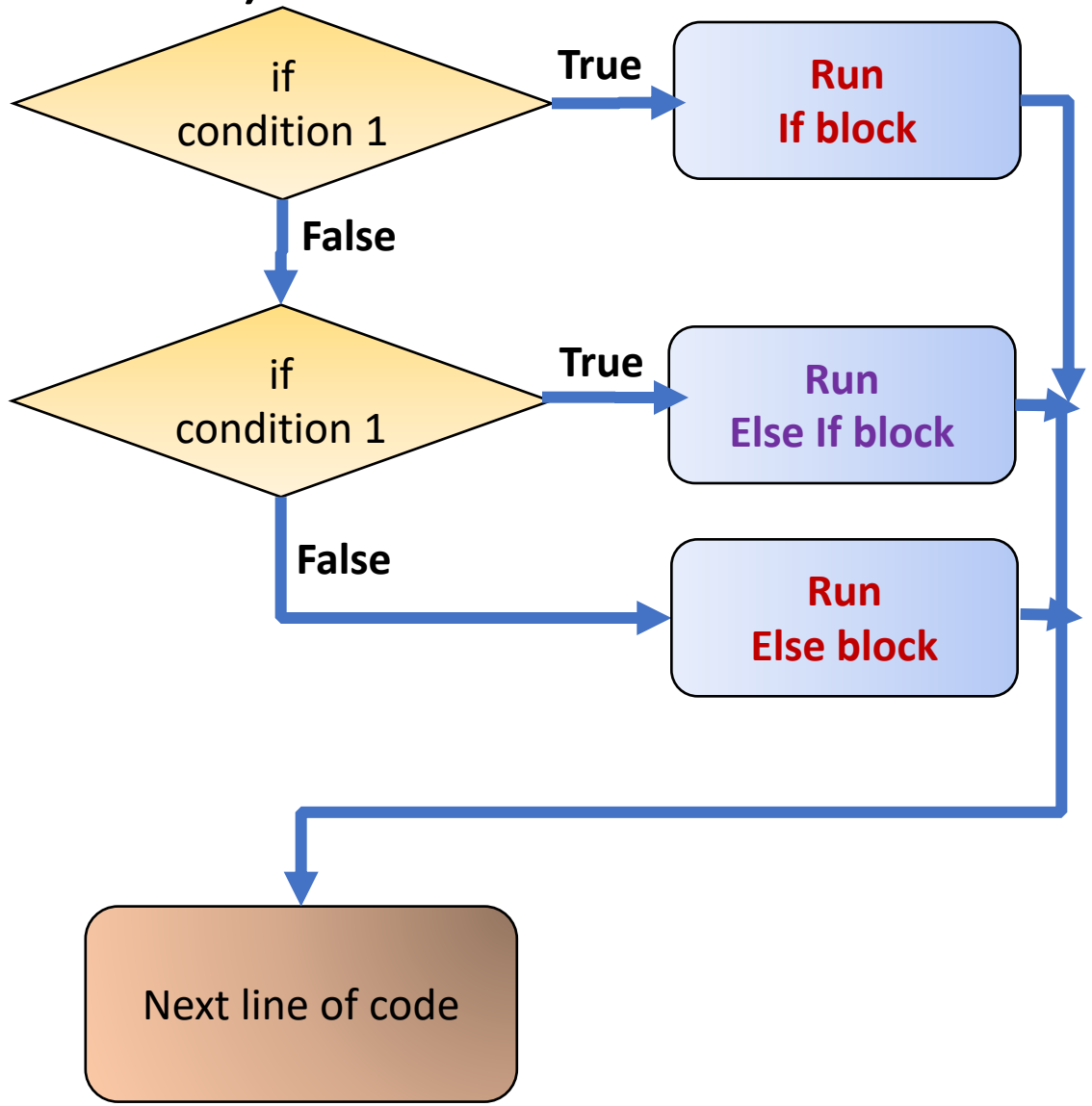


```
x=100

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)