SE103 Programming Logic Thinking

If Statements

Agenda

- Selection structure
- How to analyze the selection problem
- Case studies

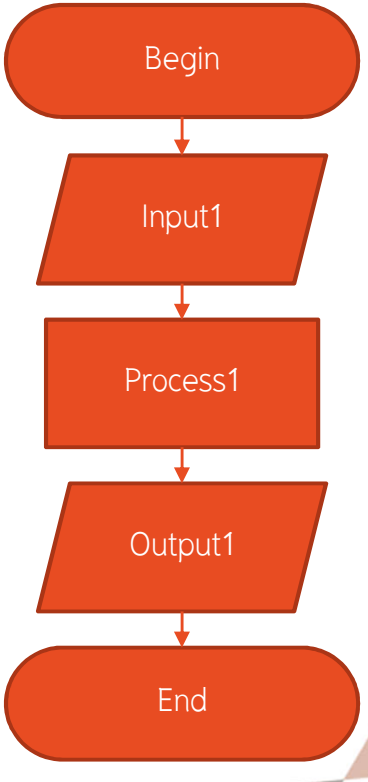
Motivation

- Sequence structure states that you will perform the same task regardless of the input

- Always execute the same sequence

of processes

- Even its produce the error



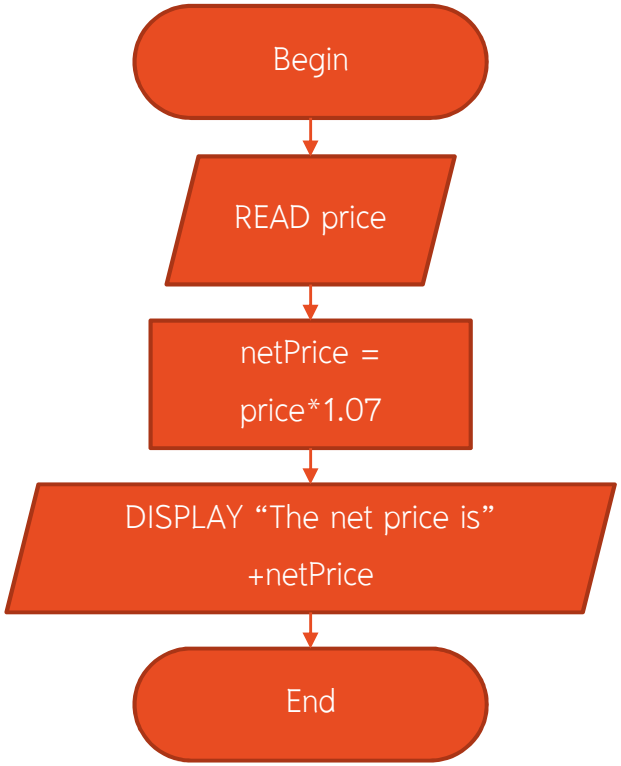
Program 1

"Create a program to calculate the net price of product. The tax is 7%."

Case study 1 : Break down the problem

- Input
  - price
- Process
  - net price = price * 1.07
- Output
  - The program will display
    - "The net price is" + net price

Case Study



price = 100

netPrice = 107

price = 200

netPrice = 214

price = –1000

netPrice = –1070

Can price be negative ?

CAMT Digital School

If statements

- Selection
  - Allow you to choose between alternatives.
  - Select the alternative based on the condition
  - Condition = Open statement (the input will decide the alternative to choose)

- Example

How do you go to university?

a) Walk If it not rains

b) Bus If it rains

To make the decision

- Identify the alternatives and its corresponding condition

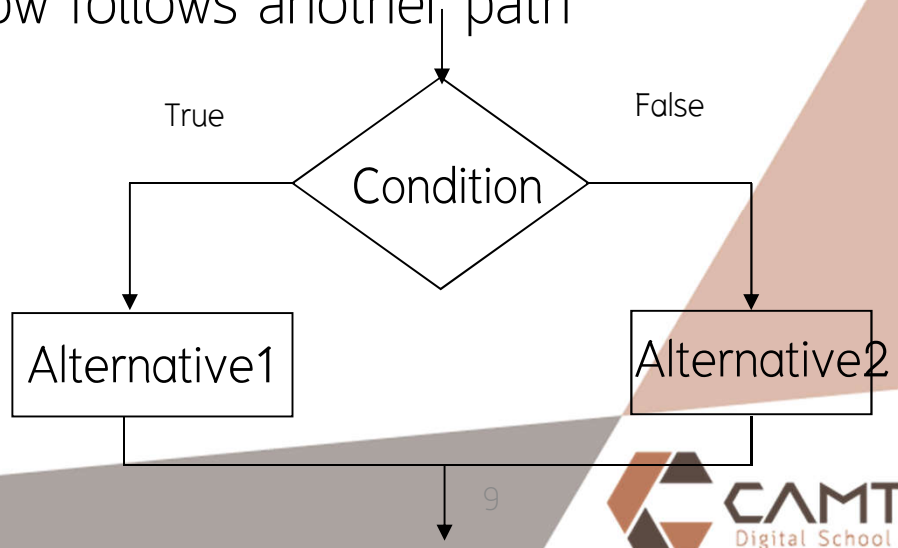
Eat at restaurant, If money > 1000	Get ‘A’, If score >= 80
Eat instant noodle, If money <= 1000	Get ‘F’, If score < 50

- Condition must be in Boolean (true, false).
  - Open statement

Flow Chart : Decision Structure

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

- The diamond, indicates a yes/no question
  - If the answer to the question is yes (True), the flow follows one path
  - If the answer is no (False), the flow follows another path



IPO and Selection structure

- Don't forget the IPO

- What is the input?
  - Identify all of the values that you need
  - Get it from user or somewhere else
- What is the process?
  - Solution to the problem
  - The selection is in this section
- What is the output?
  - Display the result
  - Make it readable

Seems easy, right?

Rewrite the problem using selection structure

"Create a program to calculate the net price of product. The tax is 7%. When the price is negative, tell the user 'The price can not be negative'."

Case study 1 : Break down the problem

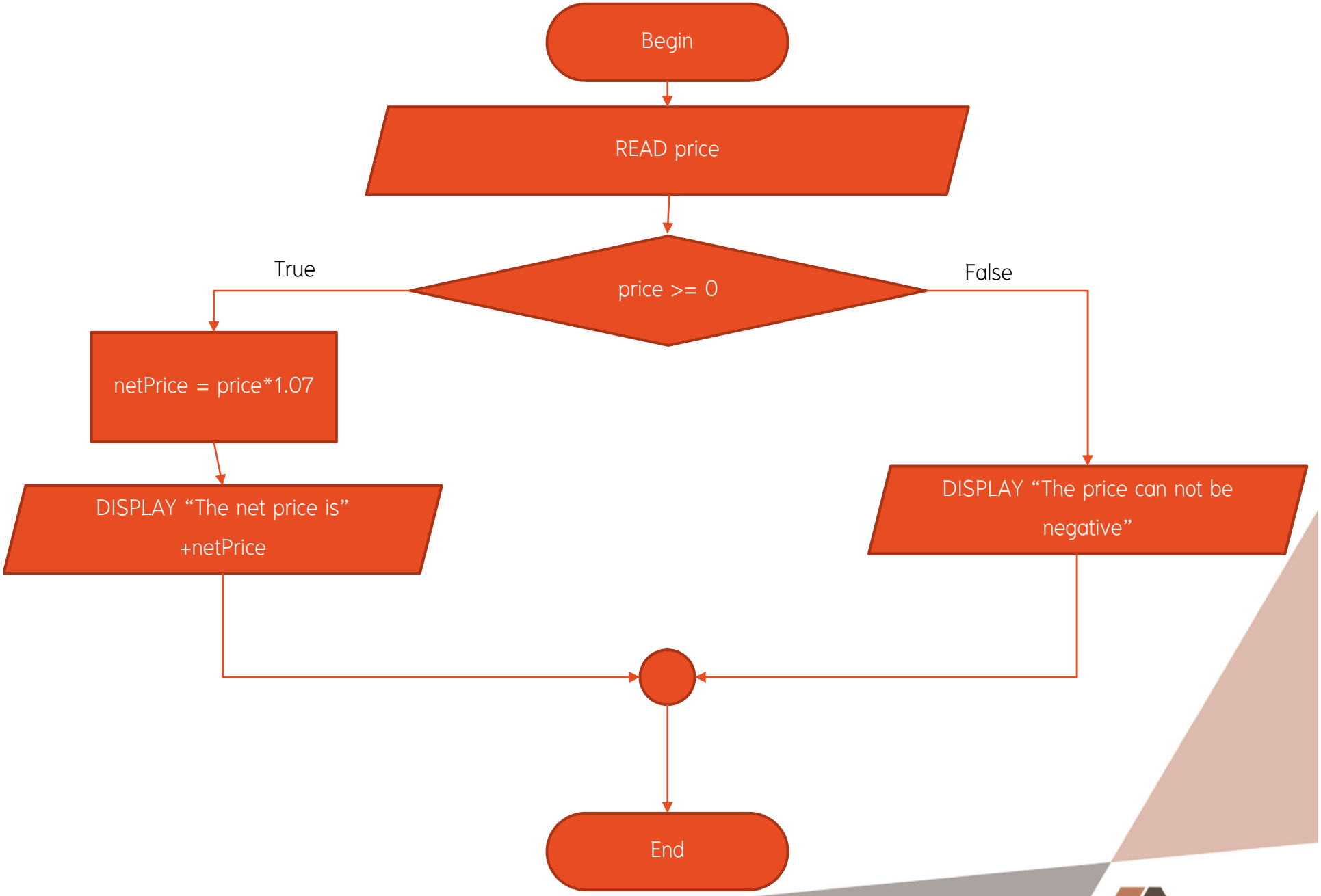
- Input
  - Price
- Process
  - If the net price is positive, then net price = price * 1.07
  - Otherwise, display “The price can not be negative”
- Output
  - The program will display
    - “The net price is” + net price

Case study 1 : Condition Analysis

- Condition
  - price must be larger than 0

    price >= 0
- Action if true
  - net price = price * 1.07
- Action if false
  - Display the warning message

    “The price can not be negative”


Begin
READ price
price >= 0
True
False
netPrice = price*1.07
DISPLAY “The net price is” +netPrice
DISPLAY “The price can not be negative”
End

Case study 2

"From the CMU regulation, students could not take the final exam if they did not attend the class more than 20%"

Case study 2 : Break down the problem

- Input
  - Number of absence
  - Total number of class
- Process
  - Calculate the number of absence in percentage.
  - The student can not take the final exam, if the number of absence is larger than 20%
  - The student can take the final exam, if the number of absence is less than 20%
- Output
  - The program will display
    - "The student can take the final exam" or
    - "The student can not take the final exam"

Case study 2 : Condition Analysis

- Condition
  - The number of absence is larger than 20%

    number of absence > 20
- Action if true
  - DISPLAY message

    "The student can take the final exam"
- Action if false
  - DISPLAY warning message

    "The student can not take the final exam"


Begin
READ numberOfAbsence, numberOfTotalClass
numberOfAbsencePercentage = numberOfAbsence*100/ numberOfTotalClass
numberOfAbsencePercentage <= 20
True
False
DISPLAY "The student can take the final exam"
DISPLAY "The student can not take the final exam"
End

Prepare a piece of paper

From another problems

- Write a program that simulates a vending machine. The machine holds three items numbered 1 through 3, with prices $1.25, $.75 and $.90, respectively. The input to your program is an integer and a floating-point number representing an item number and a sum of paid money. If the money is enough to buy the item, your program should return :

  "Thank you for buying item *X*. Your change is *Y*."

- If the money inserted is insufficient, then your program should say so.

  "Please insert another Z"

Case #1

- Enter an item number : 3
- Input money : 1.00

Output : Thank you for buying item 3. Your change is 0.10

Case #2

- Enter an item number : 1
- Input money : 1.00

Output : Please insert another 0.25

Q&A