BINUS UNIVERSITY

Subject : COMP6047
ALGORITHM AND PROGRAMMING

Year : 2019

Algorithm & Programming

People
Innovation
Excellence

People
Innovation
Excellence

Learning Outcomes

LO-1 : Explain About algorithm and its representation

- Flowchart
- Pseudocode

People
Innovation
Excellence

Outline

- Algorithm definition
- Flowchart
- Pseudocode

Algorithm Definition

- Algorithm is a procedure for solving a problem in terms of the **actions** to be executed, and the **order** in which these actions are to be executed
- We have a sequential list of actions from start/initial state to goal
- We have input, we have output.
- You will have to design and structure these list of actions.
- Derived from the word **algoris** and **ritmis**. Introduced by **Al-Khowarizmi**.
- In the programming domain, algorithm define as method that consist of structured steps in problem solving using computer.

Simple Algorithm Example

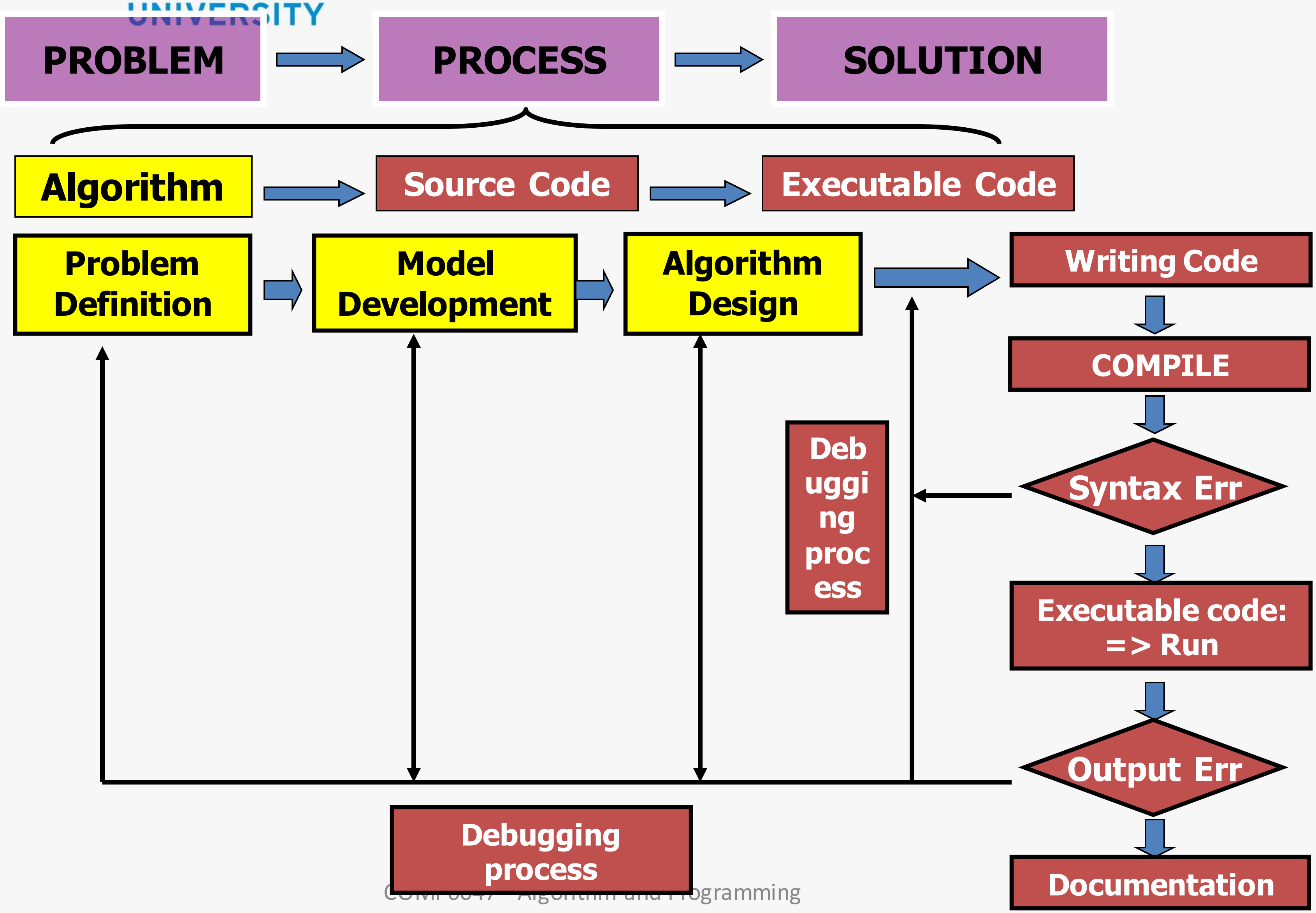
Rise and Shine Algorithm

- **Set actions**
- **Sequential order of actions**

(1)Get out of bed
(2)Take off pajamas
(3)Take a shower
(4)Get dressed
(5)Eat breakfast
(6)Carpool to work

Algorithm Development Steps

BINUS UNIVERSITY



PROBLEM

PROCESS

SOLUTION

Algorithm

Source Code

Executable Code

Problem Definition

Model Development

Algorithm Design

Writing Code

COMPILE

Syntax Err

Executable code: => Run

Output Err

Documentation

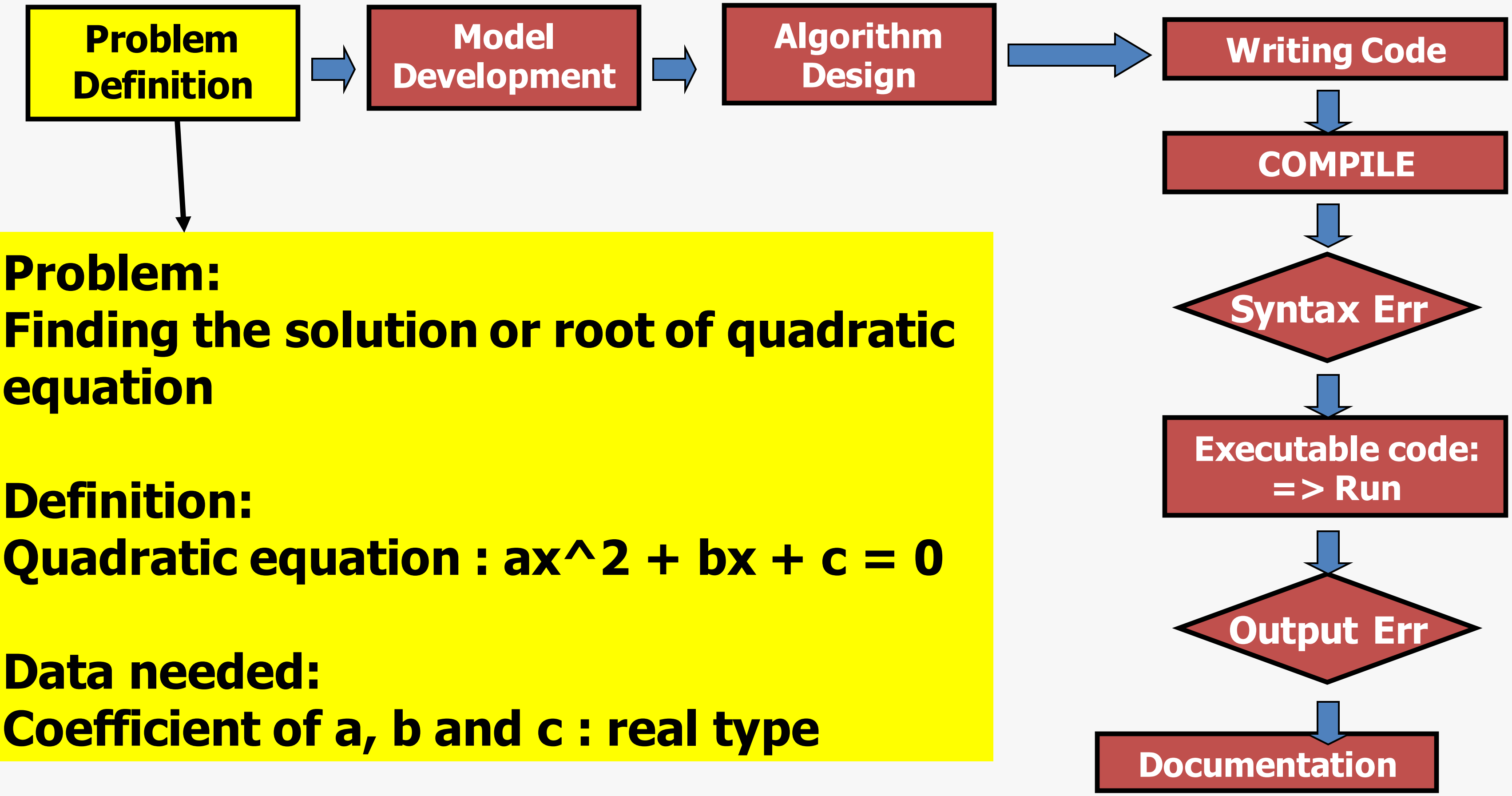
Debugging process

Debugging process

People
Innovation
Excellence

BINUS UNIVERSITY

Algorithm Development Steps



Problem Definition

Model Development

Algorithm Design

Writing Code

COMPILE

Syntax Err

Executable code: => Run

Output Err

Documentation

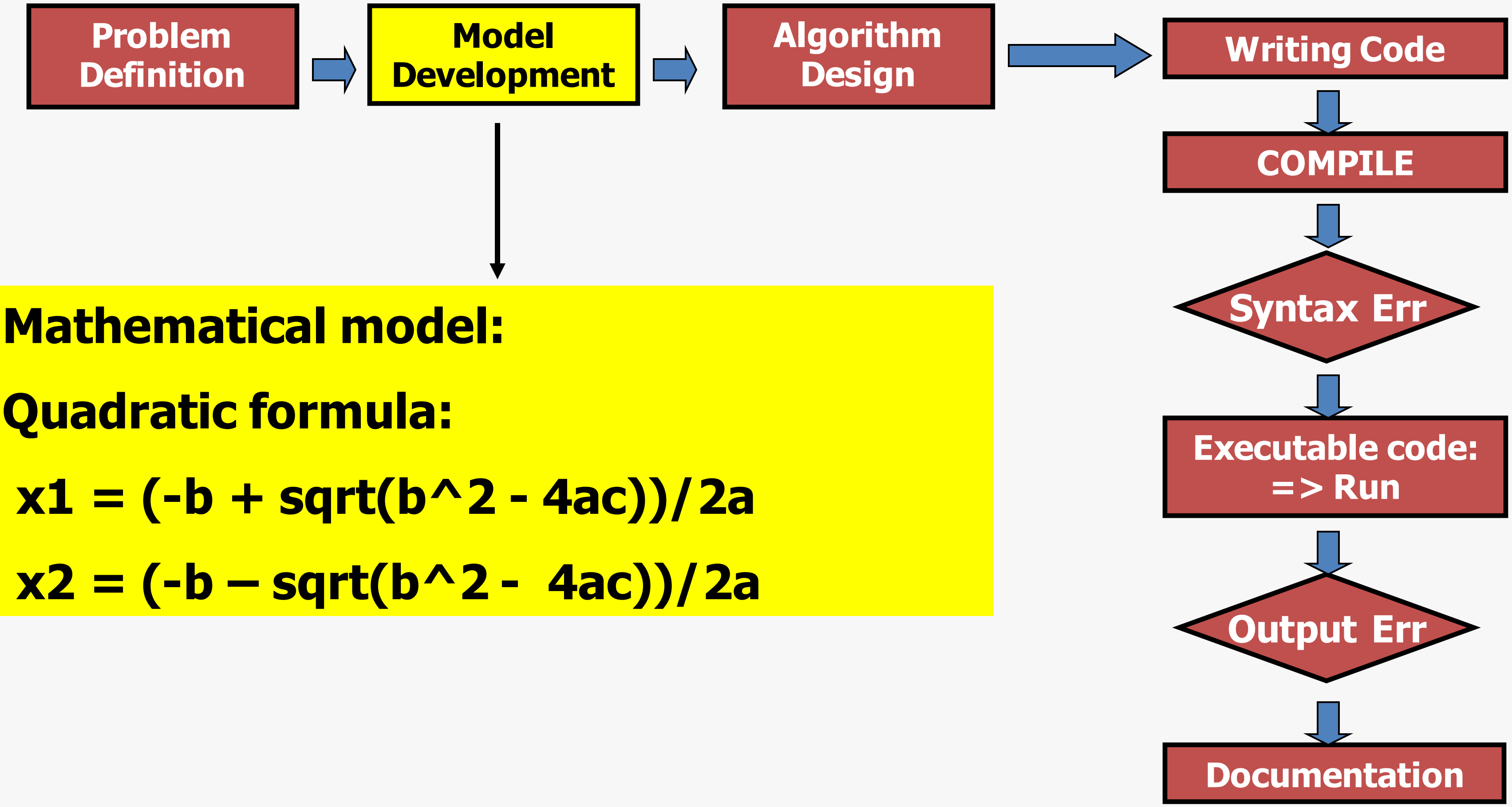
Problem:
Finding the solution or root of quadratic equation

Definition:
Quadratic equation : ax^2 + bx + c = 0

Data needed:
Coefficient of a, b and c : real type

People
Innovation
Excellence

Algorithm Development Steps



Problem Definition

Model Development

Algorithm Design

Writing Code

COMPILE

Syntax Err

Executable code: => Run

Output Err

Documentation

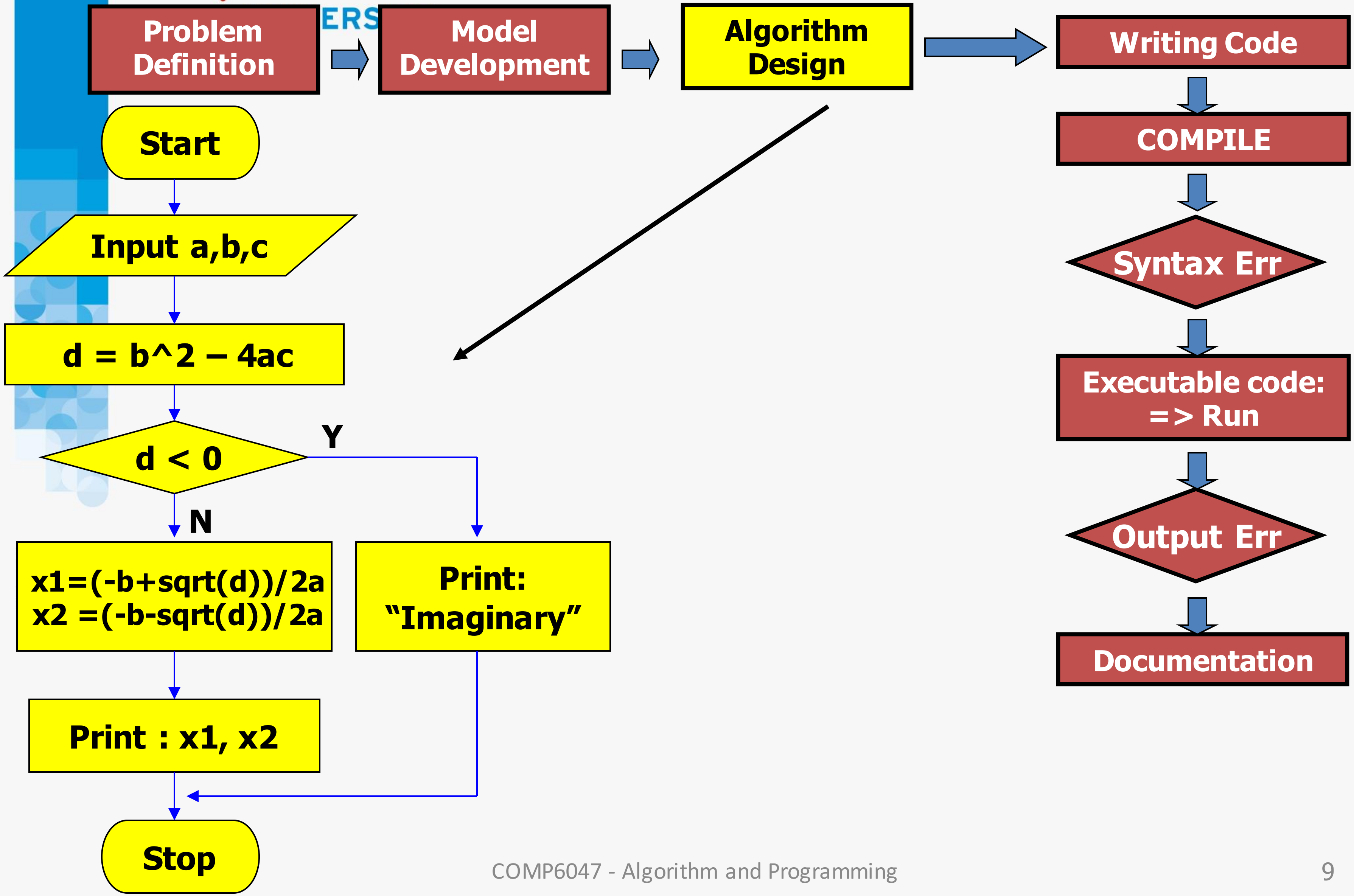
Mathematical model:

Quadratic formula:

x1 = (-b + sqrt(b^2 - 4ac))/2a

x2 = (-b – sqrt(b^2 - 4ac))/2a

Algorithm Development Steps



Problem Definition

Model Development

Algorithm Design

Writing Code

COMPILE

Syntax Err

Executable code: => Run

Output Err

Documentation

Start

Input a,b,c

d = b^2 – 4ac

d < 0

Y

N

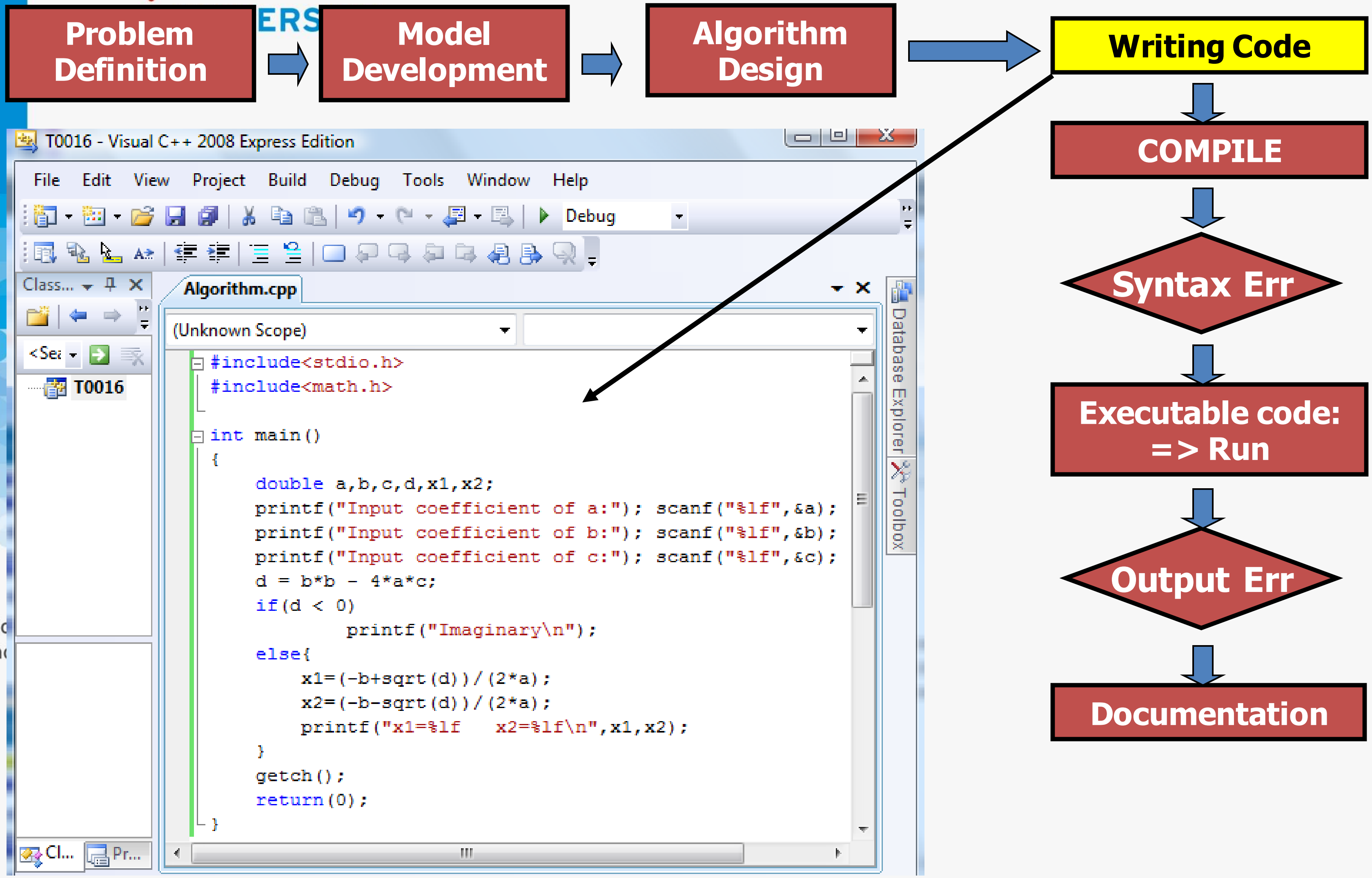
x1=(-b+sqrt(d))/2a
x2 =(-b-sqrt(d))/2a

Print: “Imaginary”

Print : x1, x2

Stop

Algorithm Development Steps

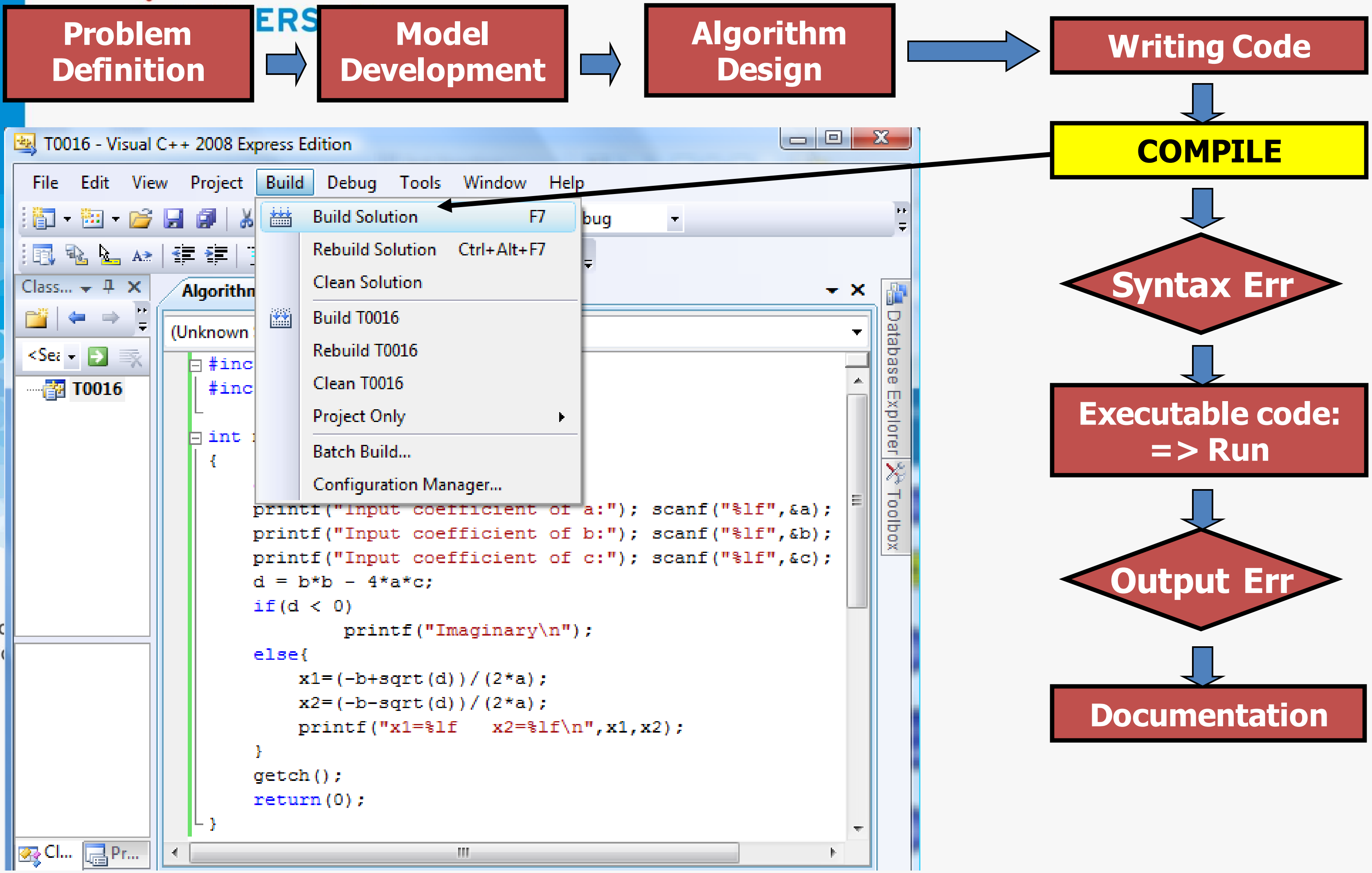


```
#include<stdio.h>
#include<math.h>

int main()
{
    double a,b,c,d,x1,x2;
    printf("Input coefficient of a:"); scanf("%lf",&a);
    printf("Input coefficient of b:"); scanf("%lf",&b);
    printf("Input coefficient of c:"); scanf("%lf",&c);
    d = b*b - 4*a*c;
    if(d < 0)
            printf("Imaginary\n");
    else{
        x1=(-b+sqrt(d))/(2*a);
        x2=(-b-sqrt(d))/(2*a);
        printf("x1=%lf   x2=%lf\n",x1,x2);
    }
    getch();
    return(0);
}
```

People
Innovatio
Excellen

Algorithm Development Steps



Problem Definition → Model Development → Algorithm Design → Writing Code → COMPILE → Syntax Err → Executable code: => Run → Output Err → Documentation

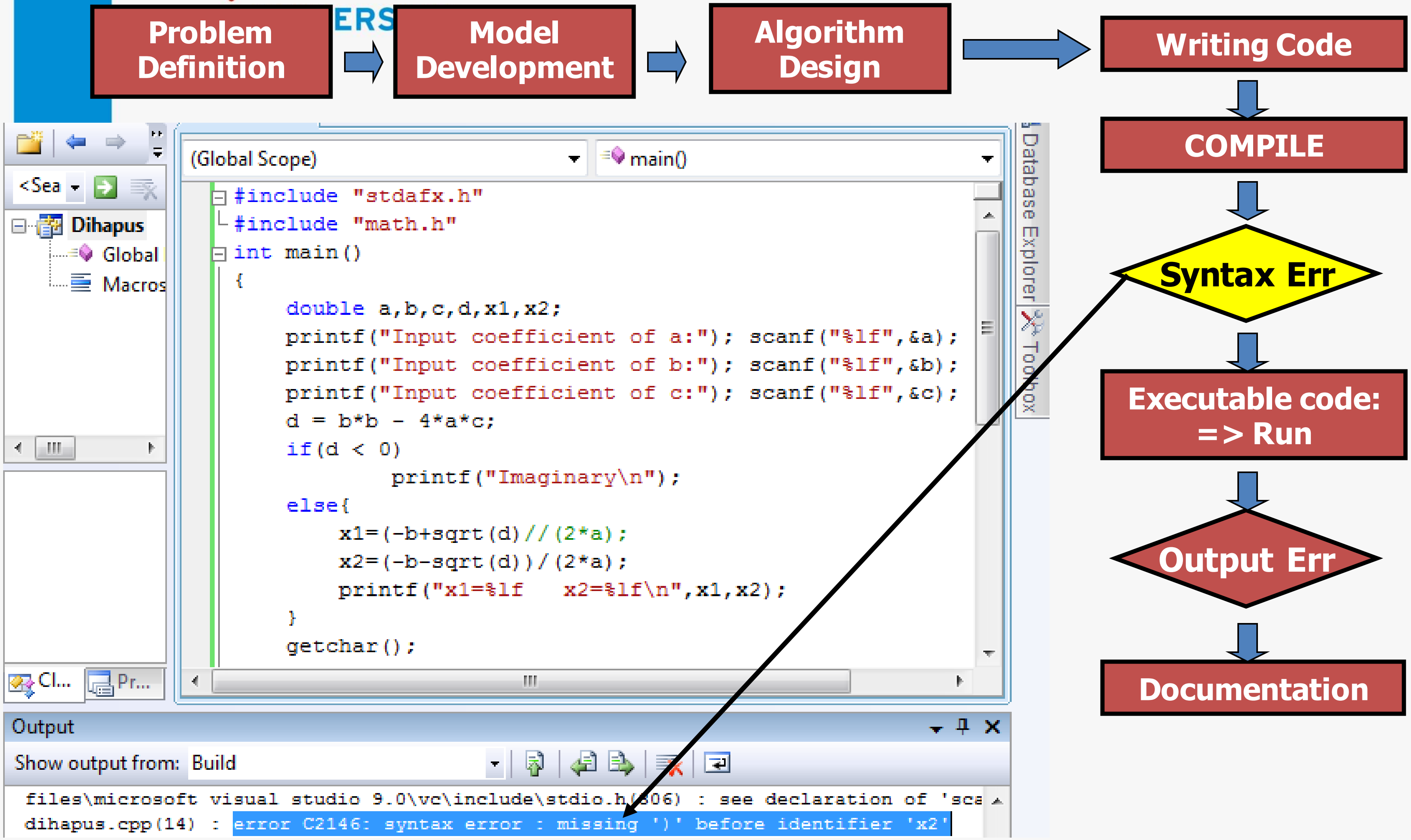
```
#inc
#inc

int
{

    printf("Input coefficient of a:"); scanf("%lf",&a);
    printf("Input coefficient of b:"); scanf("%lf",&b);
    printf("Input coefficient of c:"); scanf("%lf",&c);
    d = b*b - 4*a*c;
    if(d < 0)
            printf("Imaginary\n");
    else{
        x1=(-b+sqrt(d))/(2*a);
        x2=(-b-sqrt(d))/(2*a);
        printf("x1=%lf   x2=%lf\n",x1,x2);
    }
    getch();
    return(0);
}
```

People
Innovatic
Excellen

Algorithm Development Steps

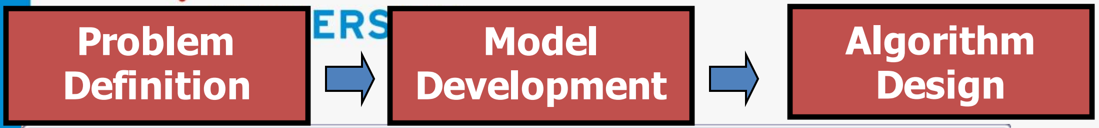


```
#include "stdafx.h"
#include "math.h"
int main()
{
    double a,b,c,d,x1,x2;
    printf("Input coefficient of a:"); scanf("%lf",&a);
    printf("Input coefficient of b:"); scanf("%lf",&b);
    printf("Input coefficient of c:"); scanf("%lf",&c);
    d = b*b - 4*a*c;
    if(d < 0)
            printf("Imaginary\n");
    else{
        x1=(-b+sqrt(d)//(2*a);
        x2=(-b-sqrt(d))/(2*a);
        printf("x1=%lf   x2=%lf\n",x1,x2);
    }
    getchar();
```

Show output from: Build

```
files\microsoft visual studio 9.0\vc\include\stdio.h(306) : see declaration of 'sca
dihapus.cpp(14) : error C2146: syntax error : missing ')' before identifier 'x2'
```

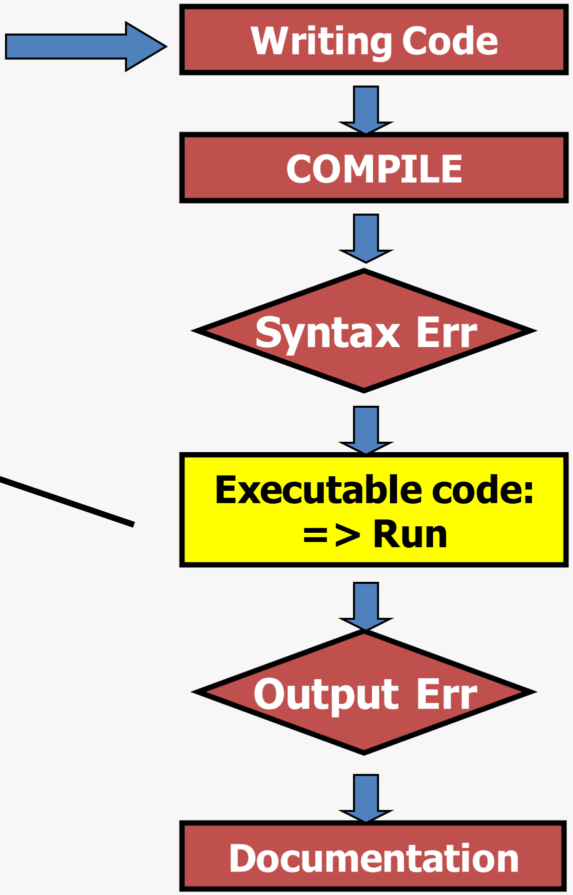
Algorithm Development Steps



Problem Definition

Model Development

Algorithm Design



Writing Code

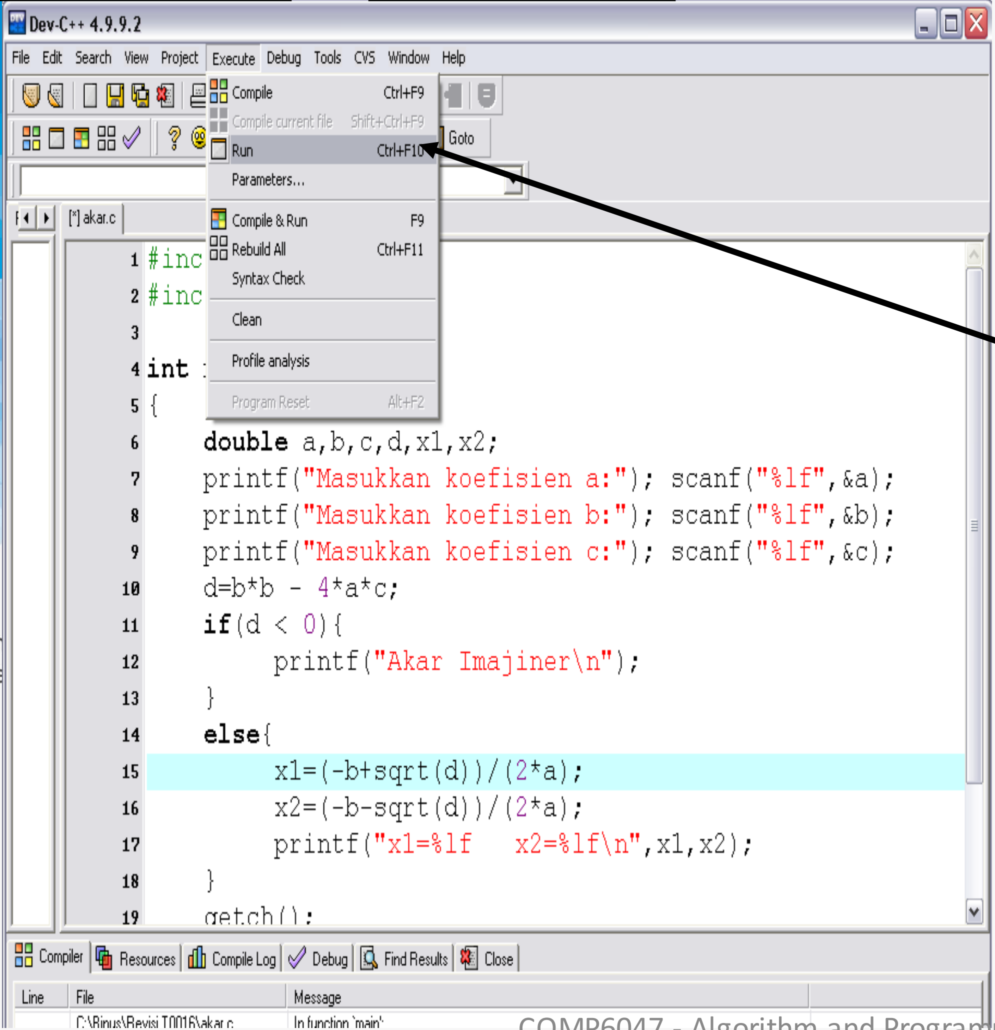
COMPILE

Syntax Err

Executable code: => Run

Output Err

Documentation

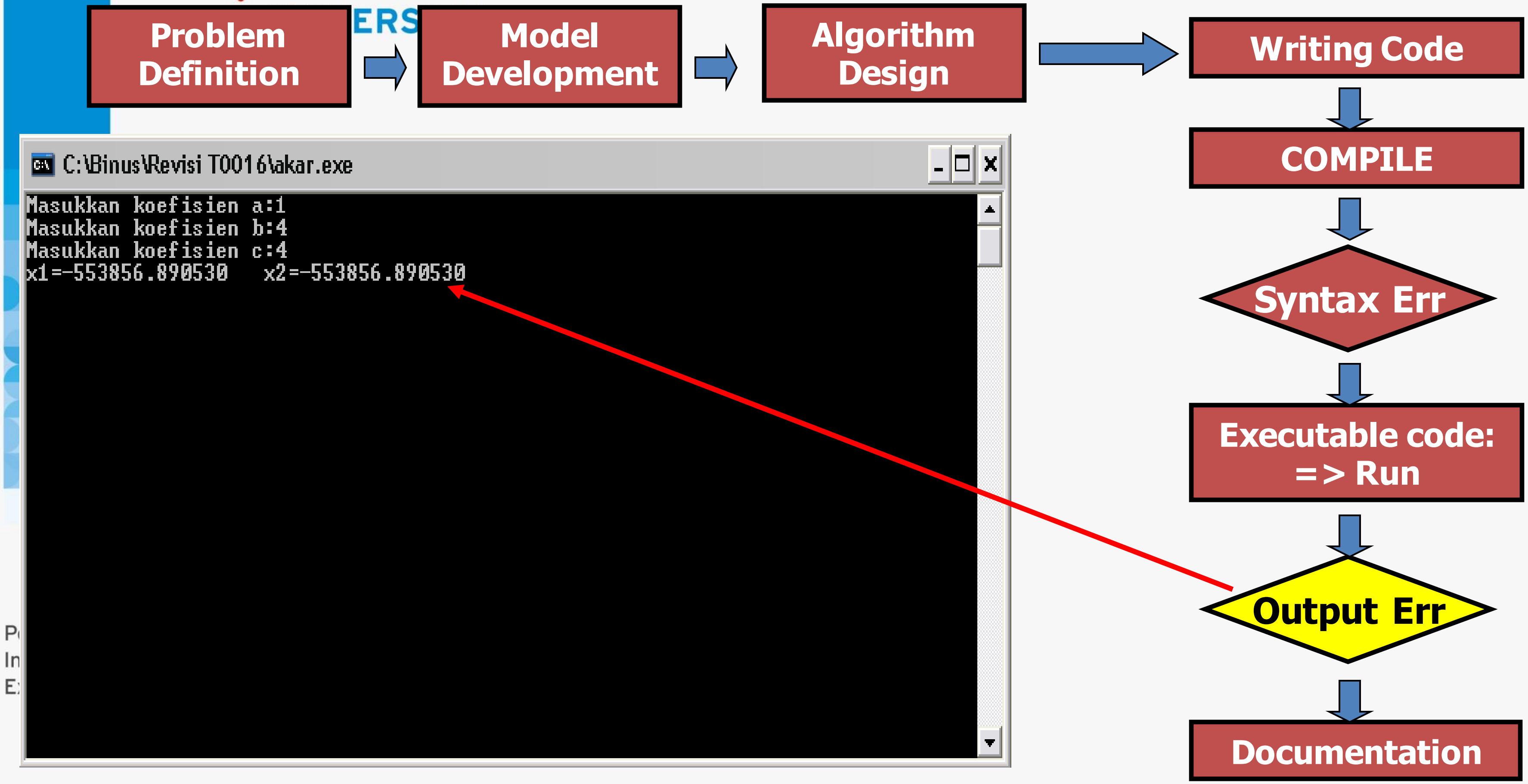


```
#inc
#inc

int :
{
    double a,b,c,d,x1,x2;
    printf("Masukkan koefisien a:"); scanf("%lf",&a);
    printf("Masukkan koefisien b:"); scanf("%lf",&b);
    printf("Masukkan koefisien c:"); scanf("%lf",&c);
    d=b*b - 4*a*c;
    if(d < 0){
        printf("Akar Imajiner\n");
    }
    else{
        x1=(-b+sqrt(d))/(2*a);
        x2=(-b-sqrt(d))/(2*a);
        printf("x1=%lf   x2=%lf\n",x1,x2);
    }
    getch();
```

Line	File	Message
	C:\Binus\Revisi T0016\akar.c	In function `main':

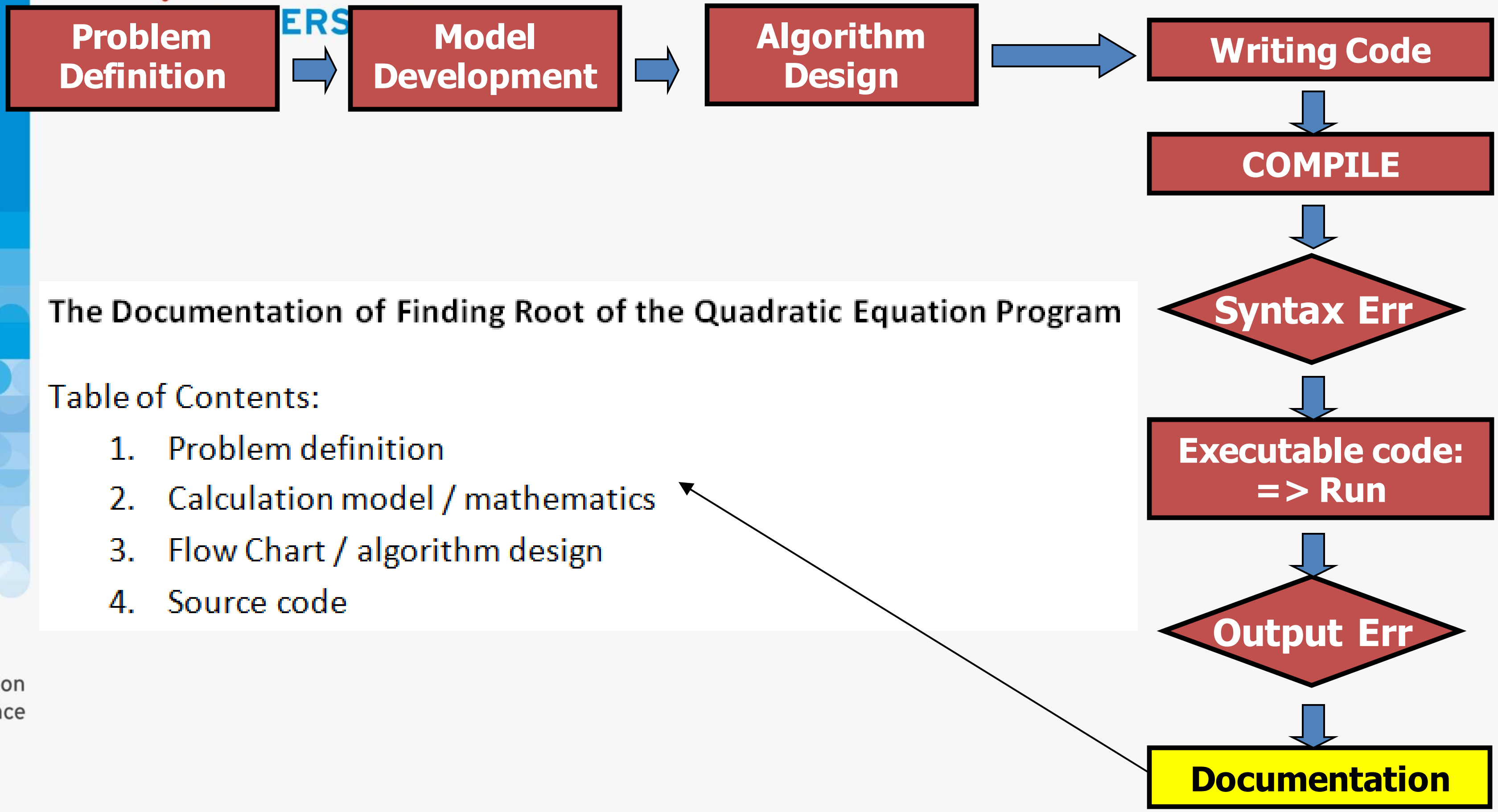
Algorithm Development Steps



```
Masukkan koefisien a:1
Masukkan koefisien b:4
Masukkan koefisien c:4
x1=-553856.890530   x2=-553856.890530
```

P
In
E

Algorithm Development Steps



The Documentation of Finding Root of the Quadratic Equation Program

Table of Contents:

1. Problem definition
2. Calculation model / mathematics
3. Flow Chart / algorithm design
4. Source code

People
Innovation
Excellence

People
Innovation
Excellence

Program Documentation

- For your 1st semester, this concept is still unimportant
- For your last semester, documentation is important in programming, especially in software engineering

Representing Algorithm

- **How to develop an algorithm?**

  We can use:
  - Writing

    Structure English (narration) and Pseudo-code.
  - Drawing

    Flow Chart

People
Innovation
Excellence

Pseudo-code

- An artificial and informal language that helps you develop algorithms
- Pseudo-code is similar to everyday English, convenient, and user friendly
- **Keywords** are used to describe control structure

  Example:

  `if, else, print, set, add, while`, etc.

People
Innovation
Excellence

Pseudo-code

Basic Computer Operation:

1. Input
2. Output
3. Compute
4. Storing value to an identifier (Store)
5. Compare (Selection)
6. Repetition (Loop)

1. Input

- Statements can be used when a computer receive information or input

  `Read, Get, Input` or `Key-In`

- Example:

```
Read bilangan
Get tax_code
Baca students_name
```

2. Output

- Statements can be used when a computer displaying information or output:

  `Print, Write, Put, Output,` or `Display`

- Example:

```
 Print "Bina Nusantara University"
 Write "Algorithm and Programming"
 Output Total
 Display image
```

3. Compute

- To do arithmetic calculation the following operators are used:

+	(add)
-	(subtract)
*	(multiply)
/	(divide)
()	(scope)

- Statement `Compute`, `Calculate` or `Add` also can be used

- Example:

```
Add number to total
Total = Total + number
```

People
Innovation
Excellence

4. Storing Value to An Identifier (Store)

- There are three ways of storing value into a variable:
  - Initializing value using statement `Initialize` or `Set`
  - Storing value as calculation result using `=`
  - To simply store a value into a variable using "`Save`" or `Store`

- Example:

```
Set Counter to 0
Total = Price * Qty
```

5. Compare

- One of the main operation in computing is comparing values and choosing options based on its result
- Keyword used: `IF`, `THEN` and `ELSE`
- Example:

```
IF Menu='1' THEN
     Discount = 0.1 * price
   ELSE
     Discount = 0.2 * price
   ENDIF
```

People
Innovation
Excellence

6. Repetition (Looping)

- To repeat an action/step, we use keyword `DOWHILE` and `ENDDO`

“tampilkan bilangan dari 0 sampai 9

- Example:

```
NUMBER = 0
(selama number < 10)
DOWHILE number < 10
  print number
  number = number +1
ENDDO
```

Pseudo-code Example

Example : Algorithm using a calculator to sum values

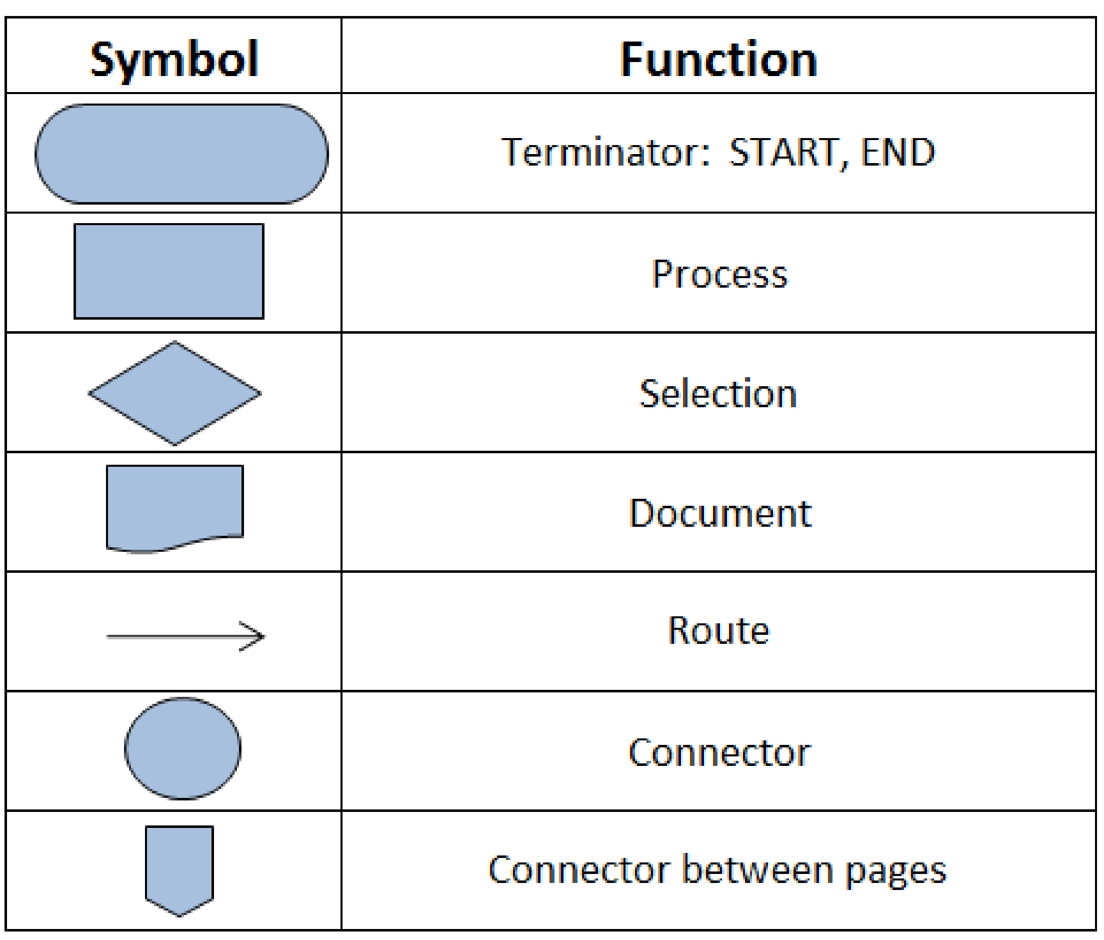
```
Start
    Set the calculator ON
    Empty any values
    Do
        Input price
        Push plus button (+)
    while all prices have been input
    print total price
    turn OFF calculator
End
```

Pseudo-code Example

Example : Algorithm to count average grade of a class

```
Start
    Set total to zero
    Set grade counter to one
    While grade counter is less than or equal to ten
            Input the next grade
            Add the grade into the total
            Add one to the grade counter
    Set the class average to the total divided by ten
    Print the class average.
End
```

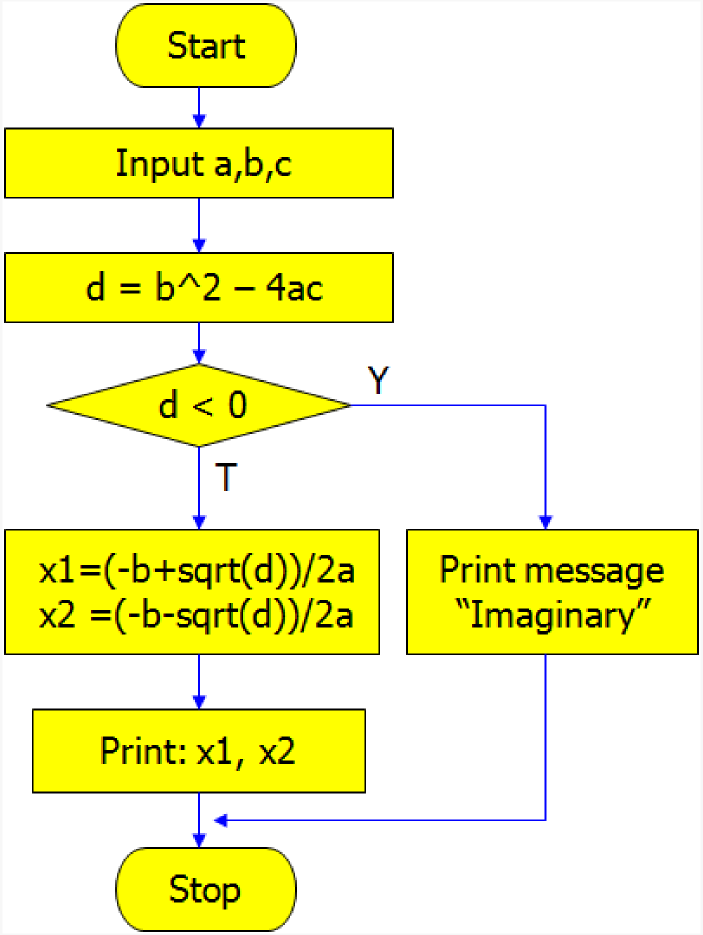
Flow Chart



Symbol	Function
⬭	Terminator: START, END
▭	Process
◇	Selection
🗎	Document
→	Route
○	Connector
⬟	Connector between pages

People
Innovation
Excellence

Flow Chart Example



People
Innovation
Excellence

Good Algorithm Practice

- Having the right logical flow to solve the problem
- Producing the correct **output** in a time efficient manner
- Written using unambiguous structured language
- Easy implementation into real programming language
- All steps and operations are clearly defined and ended

Structure Theorem

Structure theorem which makes the computer programming possible using only three control structure, which are:

1. Sequence
2. Selection
3. Repetition

People
Innovation
Excellence

1. Sequence

- Sequence is series of consecutive commands/statements
- Commonly programming language has sequence of statements flowing from top of the program to its end

People
Innovation
Excellence

1. Sequence

- Example :

```
Print “Number of students:”
Set total to 49
Print “Add new student:”
Read newStudent
total = total + newStudent
Print “Number of students:”
Print total
```

- Description

  Sequence of command is from the 1st line to the end of code. If newStudent input is 2 then total that later on printed out is 51

People
Innovation
Excellence

2. Selection

- Selection control structure is structure that allow us to choose from several options of statement/command
- The first statement will be executed **if** the condition is satisfied, if not then the **else** statement will be executed (if the other exist)

2. Selection

- Example :

```
IF Day=1 THEN
  Print "Monday"
ELSE
  Print "Obviously not Monday"
```

- Description

  The word “Monday” will be printed out if Day’s value equal to **1**, else it will print out the sentence “Obviously not Monday”.

People
Innovation
Excellence

3. Repetition

- A number of statements/commands can be repeated several times using Repetition structure control

- Statements/commands will be repeated while the looping condition is satisfied

  (may use DOWHILE – ENDDO)

People
Innovation
Excellence

3. Repetition

- Example :

```
Stars = 0
DOWHILE Stars < 5
  Print Stars
   Stars = Stars + 1
ENDDO
```

- Description

  At first Stars' value equals to 0, after following the DOWHILE looping Stars' value will be updated 5 times resulting:

  0 1 2 3 4

BINUS UNIVERSITY

Exercise

1. Using the **Pseudo-code or english**, create:
   a. an algorithm to calculate a rectangle area!
   b. an algorithm to change second into hour and minute unit!
   c. an algorithm to decide whether an input number is an odd or even number!
   d. an algorithm to calculate a circle area!
   e. an algorithm to accept three numbers and find the max number!
2. Repeat no. 1 using **Flow Chart**

People
Innovation
Excellence

- E
  - Input a,b,c
  - Jika b > a maka cek c > b, jika iya maka print c, jika tidak print b
  - a > b ? Cek c > a. Jika iya print c, jika tidak print a.

- A.
  - Input panjang dan lebar
  - Luas = panjang * lebar
  - Print luas

People
Innovation
Excellence

People
Innovation
Excellence

- D
  - Model math = luas lingkaran pi*r*r
  - Input r
  - Pi = 3,14 atau 22/7
  - Luas = Pi * r * r
  - Print Luas

- B
  - Input detik
  - Menit = detik/60
  - Jam = detik/3600
  - Print menit
  - Print jam

- C
  - Jika angka habis dibagi 2 = 0 hasilnya genap, kalau tidak ganjil
  - Input angka
  - Jika angka habis dibagi 2 = 0, print genap, kalau tidak (else) print ganjil

People
Innovation
Excellence

People
Innovation
Excellence

Summary

- Algorithm is a procedure for solving a problem in terms of the actions to be executed
- Algorithm development steps consists of: problem definition, model development, algorithm design, writing code, and documentation
- We can use writing (Structure English and Pseudo-code) or drawing (Flow Chart) to represent algorithm
- Basic Computer Operation: input, output, compute, store, compare, and repetition (loop)
- Structure theorem are sequence, selection, and repetition

References

- Tonny Gaddis. (2018). Starting Out with Programming Logic and Design. Pearson. New York
- Pseudocode Examples: http://www.unf.edu/~broggio/cop2221/2221pseu.htm
- Computer & Internet Help : Understanding Flowchart Symbols: http://www.youtube.com/watch?v=xLoL7tlJYws

People
Innovation
Excellence

People
Innovation
Excellence

END