

Problem Solving

Problem Solving Concept for Computer Science

by

Noor Azida Binti Sahabudin
Faculty of Computer Systems & Software Engineering
azida@ump.edu.my



OER Problem Solving by Noor Azida Binti Sahabudin work is under licensed [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Chapter Description

- **Aims**

- Differentiate between **constant** and **variable**
- Understand and able to use **various data types**
- Understand and able to use **functions**
- Understand and able to differentiate between **operators, operand and resultants**
- Able to set up and evaluate **expression and equation** using variables, constants, operators and the hierarchy of operations.

- **References**

- Sprankle, M., and Hubbard, J., (2012). Problem Solving and Programming Concepts : 9th Edition. Prentice Hall, 2012. ISBN : 0132492644



Constant & Variable

CONSTANT	VARIABLE
A specific alphabetical and/or numeric value that never changes during the processing of all the instructions in a solution	The variable name should be consistent , but the value of the variable may change during processing
Named constants rules: <ol style="list-style-type: none">1. A constant cannot be changed after it is initially give a value2. Storage location given a name3. Referred to by the given name	Rules: <ol style="list-style-type: none">1. Values of the contents for name variable location can be changed2. Storage locations are given names3. Referred to by variable name in the instructions
Example: PI Value 3.124	Example: Variable name – Height Value 165.2

(Sprankle and Hubbard, 2012)

Example of Constant

```
#include <stdio.h>
#define MONDAY 1
#define TUESDAY 2
#define WEDNESDAY 3
#define THURSDAY 4
#define FRIDAY 5
#define SATURDAY 6
#define SUNDAY 7
const float pi = 3.1415;
const int id_no = 12345;
```

```
int main(void)
{
    int today = SATURDAY;

    if ((today == SATURDAY) || (today == SUNDAY))
    {
        printf("Weekend\n");
    }
    else
    {
        printf("Go to work or school\n");
    }
    return 0;
}
```

A constant cannot be changed after it is initially given a value

Example of Variable



Variable have two parts which are variable name and variable value.
Example in the following table

Variable name	Value
Sales	2512.50
Weight	53.55
City	"Pahang"
ZipCode	"25300"

Rules for Naming and Using Variables

Must according to what it represent

Example:

StaffId for ID staff number,
StaffPhoneNum for staff phone
number

Short name as possible but clearly

Example:

FthrNm for father name (not
clear)
CN for contact number (not clear)

Do not use space

Example:

Staff Address (wrong)
StaffAddress (right)
Staff_Address (right)

Do not use a dash, any other
mathematical operator symbol)

Example:

StdN-Name (wrong)
StdNName (right)

Must start with letter

Example:

6%GST (wrong)
#VendorId (wrong)

Do not use symbol

Example:

Client@email (wrong)
Staff&Name (wrong)

Use exact variable name in all
places where the data item is used

Be consistent, when using upper
or lowercase character

Example:

Name ≠ name
CompanyAdd ≠ CompanyADD

Example of constant and variable

CASE STUDY:

Students first year register subject in Faculty System Computer & Software Engineering. What possible **data item** for constant and variable in this situation?



Example of constant and variable

CASE STUDY:

Students first year register subject in Faculty System Computer & Software Engineering. What possible **data item** for constant and variable in this situation?

ANSWER

CONSTANT

VALUE NOT CHANGE

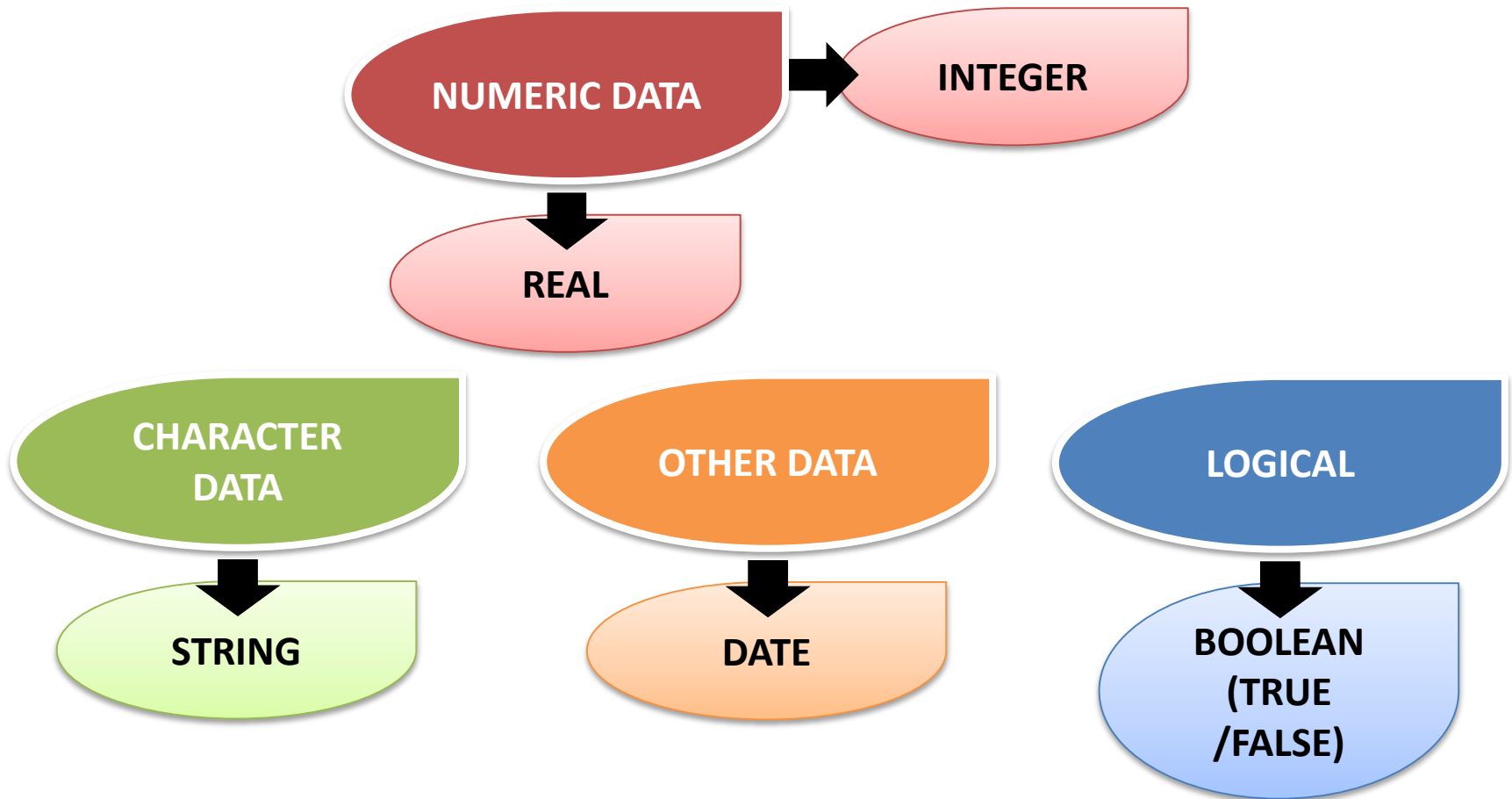
- CODE SUBJECT
- SUBJECT NAME
- SUBJECT CREDIT HOUR

VARIABLES

VALUES CHANGE

- LECTURER NAME
- SECTION NUMBER

Data Types



Exercise 1: Constant and variable

Identify between constant & variable

Value	Constant / Variable
Staff_name	
Num_of_day_in_August	
List_of_Month	
Commission	
Max_value	
Date	

Data

Definition of DATA:

Data are unorganized facts insert into computer known as INPUT (process by program) returned to user as OUTPUT or information (Sprankle & Hubbard, 2012)

Data uses in computer are many different types and computer must be told the data type of each variable or constatnt (Sprankle & Hubbard, 2012)

Data Types

Numeric data:

Data type that **can be used
in calculations**

Integer numbers

All whole numbers.

Example: Quantity of an item 0, 101, -54

Real numbers

Floating point numbers. Example 5.31×10^3 , 7.31×2^{-1}

Whole number with decimal parts. Example: Price of an item 3.21, 12.89

Data Types

Not use for calculation

Character data:

- All letters, numbers and special symbols
- Can not be use for calculation
- Surrounded by quotation marks
- Example: "A" "7" "(" "=" "%"

Character String data:

- Combinations of **more than one character**
- Surrounded by quotation marks
- Example:
- Company Name: "ABC Sdn Bhd"
- Account Number: "AW923310"
- Zipcode: "28000"
- Phone No: "011-2224409"

Extra notes: Concatenation

- Joined together (+ operator)
- Example: "4" + "4" = "44" (not "8")

Data Types

Logical data:

- Use for YES or NO decisions
- TRUE or FALSE
- Example: Are you absent today? True (Yes), False (No)

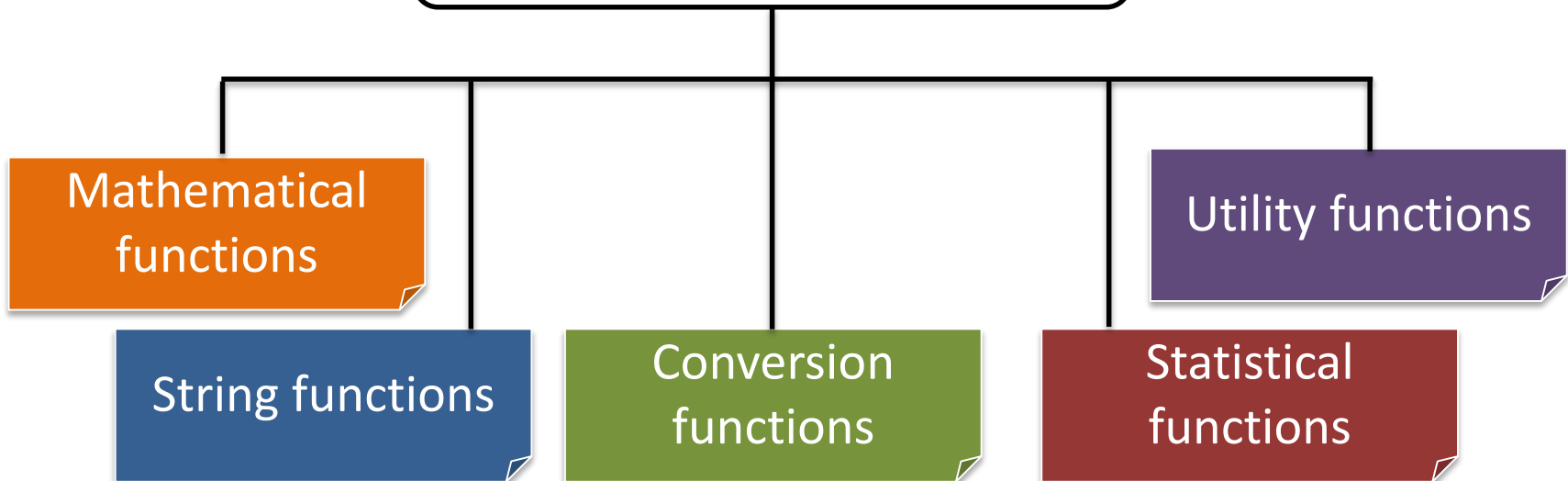
Exercise 2: Identify Data Type

Data Item	Variable Name	Data Type
Company Address	CompanyAdd	Character String
Your Age		
Date		
Price		
Identity Card Number		
Quantity		
Quality		

Functions

Functions are small sets of instructions that perform specific tasks and return values (Sprankle and Hubbard, 2012).

Basic functions



Mathematical Functions

Function	Definition	Example	Result
<i>Sqrt (N)</i>	Returns the square root of <i>N</i>	<i>Sqrt (64)</i>	8
<i>Abs (N)</i>	Returns the absolute value of <i>N</i>	<i>Abs (-11)</i>	11
<i>Round (N, n1)</i>	Returns the rounded value of <i>N</i> to the <i>n1</i> place	<i>Round (1.1993, 3)</i>	1.119
<i>Integer (N)</i>	Returns the closest whole number less than or equal to <i>N</i>	<i>Integer (9.8976)</i>	9
<i>Random</i>	Returns a random number between 0 and 1 but not 1		
<i>Sign (N)</i>	Returns the sign of <i>N</i> : 1 when <i>N</i> is +ve, 0 when <i>N</i> is zero, -1 when <i>N</i> is -ve	<i>Sign (-6.7)</i>	-1

Conversion, Statistical, Utility Functions

Function	Definition	Example	Result
Conversion functions			
<i>Value (S)</i>	Returns a string value into a numeric value	<i>Value ("65.91")</i>	+65.91
<i>String (N)</i>	Changes a numeric value into a string value	<i>String (+65.91)</i>	"65.91"
Statistical function			
<i>Average (list)</i>	Returns the average of a list of numbers	<i>Average (1, 9, 32, 8)</i>	12.5
<i>Max (list)</i>	Returns the maximum value from a list of numbers	<i>Max (1, 9, 32, 8)</i>	32
<i>Min (list)</i>	Returns the minimum value from a list of numbers	<i>Min (1, 9, 32, 8)</i>	1
<i>Sum (list)</i>	Returns the sum of a list of number	<i>Sum (1, 9, 32, 8)</i>	50
Utility functions			
<i>Date</i>	Returns the current date from the system	Date	20/06/85
<i>Time</i>	Returns the current time from the system	Time	10: 55: 21
<i>Error</i>	Returns control to the program when a systems error occurs		

String Functions

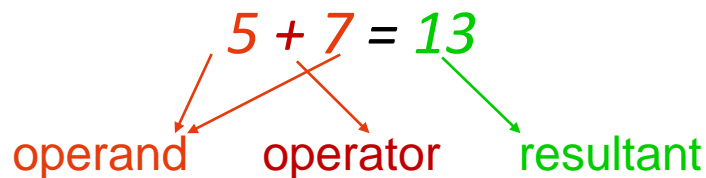
Function	Definition	Example	Result
<i>Mid (S, n1, n2)</i>	Returns a set of n2 characters starting at n1 in the string S	<i>Mid (S, 3, 2)</i> Where S = "Welcome"	lc
<i>Left (S, n)</i>	Returns a set of n character of the left side of the string S	<i>Left (S, 3)</i> Where S = "Welcome"	Wel
<i>Right (S, n)</i>	Returns a set of n characters on the right side of the string S	<i>Right (S, 3)</i> Where S = "Welcome"	ome
<i>Length (S)</i>	Returns the number of characters in the string S	<i>Length (S)</i> Where S = "Welcome"	7

Operators

According to Sprankle and Hubbard, (2012):

1. Operators are the data connectors with **expression** and **equations**.
2. Two concepts related to the operator is **operand** and **resultant**.
 - **Operand** : data that the operator connects and processes
 - **Resultant** : answer that result when the operation is completed

Example :



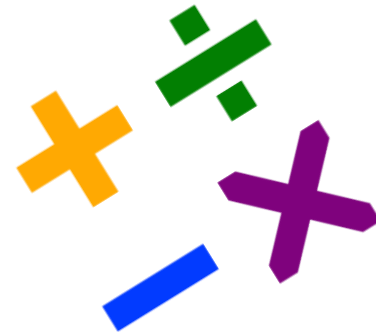
Operators

Types of Operators

Mathematical
Operators

Relational
Operators

Logical
Operators



Mathematical Operators

Operator	Computer symbol	Operation	Resultant
Addition	+	$4.1 + 2.2$	6.3
Subtraction	-	$6.5 - 4.5$	2.0
Multiplication	*	$2.0 * 6.0$	12.0
Division	/	$7.0 / 2.0$	3.5
Integer division	\	$13 \setminus 2$	6
Modulo Division	MOD	$13 \text{ MOD } 2$	1
Power	^	4^2	16

Relational Operators

Operator	Computer Symbol	Operation	Resultant
Equal to	=	$5 = 7$	FALSE
Less than	<	$5 < 7$	TRUE
Greater than	>	$5 > 7$	FALSE
Less than or equal to	<=	$5 <= 7$	TRUE
Greater than or equal to	>=	$5 >= 7$	FALSE
Not equal to	<>	$5 <> 7$	TRUE

Logical Operators

Operator	Computer Symbol	Operation	Resultant
Not	NOT	NOT TRUE	FALSE
And	AND	TRUE AND TRUE	TRUE
Or	OR	TRUE OR FALSE	TRUE

Logical Operators

NOT	Operand	Result
	TRUE	FALSE
	FALSE	TRUE

AND	Operand 1	Operand 2	Result
	TRUE	TRUE	TRUE
	TRUE	FALSE	FALSE
	FALSE	TRUE	FALSE
	FALSE	FALSE	FALSE

OR	Operand 1	Operand 2	Result
	TRUE	TRUE	TRUE
	TRUE	FALSE	TRUE
	FALSE	TRUE	TRUE
	FALSE	FALSE	FALSE

Hierarchy of Operations

Order of Operations	Operand Data Type	Resultant Data Type
() reorders the hierarchy ; all operations are completed within the parenthesis using the same hierarchy.		
1. Functions		
Mathematical Operations		
2. Power	Numeric	Numeric
3. \ , MOD	Numeric	Numeric
4. * , /	Numeric	Numeric
5. + , -	Numeric	Numeric
Rational Operator		
6. = , < , > , <= , >= , <>	Numeric or String or Character	Logical (True or False)
Logical Operator		
7. NOT	Logical (True or False)	Logical (True or False)
8. AND	Logical (True or False)	Logical (True or False)
9. OR	Logical (True or False)	Logical (True or False)

Exercise 3

1. Solve following equation

i. $K = 2 + (5 / 9 * 3 + 2) * 3 \setminus 5$

ii. $L = (2 + 5 / 9) * 3 + 2 * (3 \setminus 5)$

iii. Given $A = 5, B = 4, C = 3, D = 12$

a. $E = A * B + D / C$

b. $E = C \text{ MOD } A * B$

c. $E = 5 * A \setminus D * (B + 1)$

d. $E = D / B * ((A + 4) \setminus (C + 1))$

Expressions and Equations

- **Expression**: Processes data, the operands through the use of operator
 - Example: $\text{length} * \text{width} * \text{height}$
- **Equation**: stores the resultant of an expression in a memory location in the computer through the equal (=) sign
 - Example: $\text{volume} = \text{length} * \text{width} * \text{height}$

(Sprankle & Hubbard, 2012)

Expressions and Equations

- **Expressions**

- Use no equal sign
- A part of an instruction/equation
- Not store the resultant

- **Equations**

- Often called as “**Assignment statement**”
- Equal sign (=) : **does not mean equals, but replaced by or is assigned the value of**

(Sprankle & Hubbard, 2012)

Expressions and Equations

Expressions	Equations
X+ Y X and Y are numeric. The resultant is numeric and is not stored	Z= X+Y Z, X and Y are numeric. The resultant is stored in Z;
X <Y X and Y are numeric, character, or string. The resultant is logical and is not stored	Z = X < Y X and Y are numeric, character or string. The resultant is stored in Z; Z is logical
X OR Y X and Y are logical The resultant is logical and is not stored	Z = X OR Y Z, X and Y are logical The resultant is stored in Z;

(Sprankle & Hubbard, 2012)

Expressions and Equations

NUMERIC EXPRESSION

MATHEMATICAL EXPRESSION

$$A(2B + 1) - \frac{B}{A - 5}$$

COMPUTER EXPRESSION

$$A * (2 * B + 1) - B / (A - 5)$$

MATHEMATICAL EQUATION

MATHEMATICAL EQUATION

$$A + 11 = B(C - 1)$$

COMPUTER EQUATION

$$A = B * (C - 1) - 11$$

RELATIONAL EXPRESSION

MATHEMATICAL EXPRESSION

A is greater and equal to B - 1

COMPUTER EXPRESSION

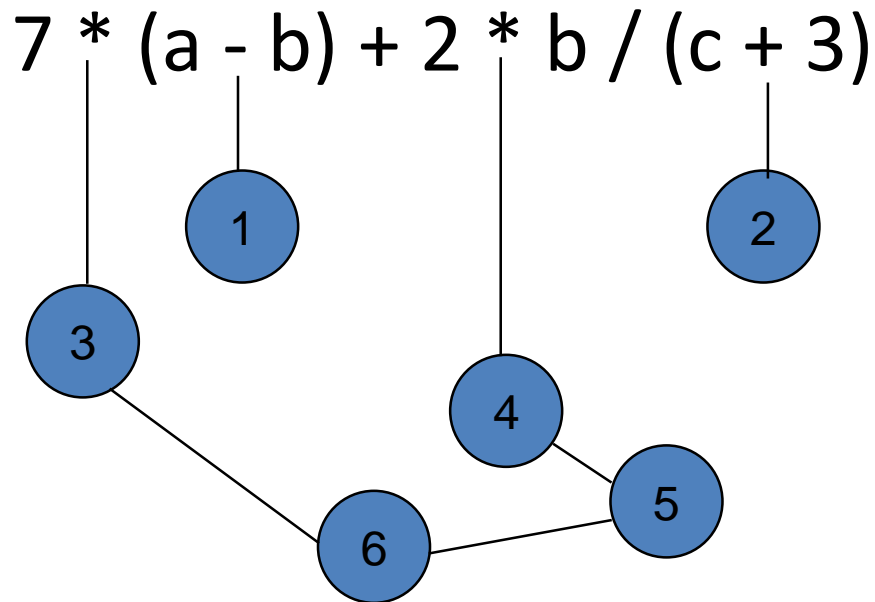
$$A \geq B - 1$$

(Sprankle & Hubbard, 2012)

Example: Mathematical Expression

Follow the hierarchy of operators

Example : $7 * (a - b) + 2 * b / (c + 3)$

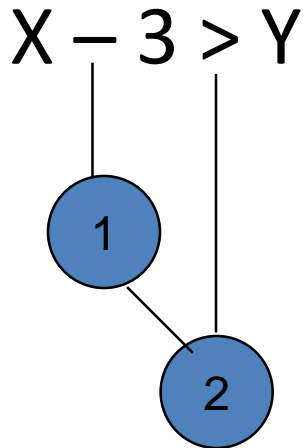


No	Operation	Resultant
1	$a - b$	3
2	$c + 3$	8
3	$7 * \text{resultant of 1}$	21
4	$2 * b$	16
5	Resultant of 4 / resultant of 2	2
6	Resultant of 3 - resultant of 5	19

Assume : $a = 11, b = 8, c = 5$

Example: Relational Expression

Example : $X - 3 > Y$; Assume $X = 5, Y = 3$



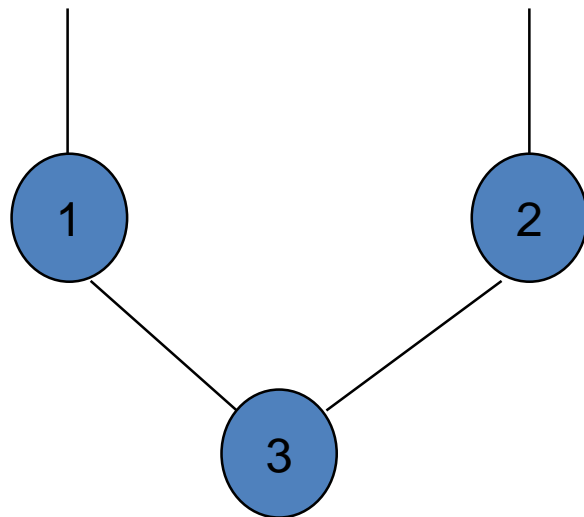
No	Operation	Resultant
1	$X - 3$	2
2	Resultant of $1 > Y$	FALSE

Example: Logical Expression

Example : $X \text{ AND } Y \text{ OR } Z \text{ AND } X$

assume ; $X = \text{FALSE}$, $Y = \text{FALSE}$, $Z = \text{TRUE}$

$X \text{ AND } Y \text{ OR } Z \text{ AND } X$

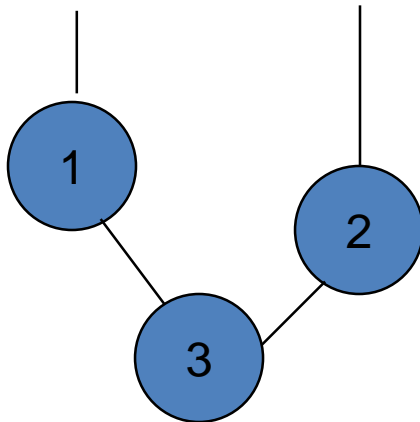


No	Operation	Resultant
1	X AND Y	FALSE
2	Z AND X	FALSE
3	Resultant of 1 OR Resultant of 2	FALSE

Example: Logical Expression

Developing a Table of All Possible Resultants of a Logical Expression

NOT A OR NOT B



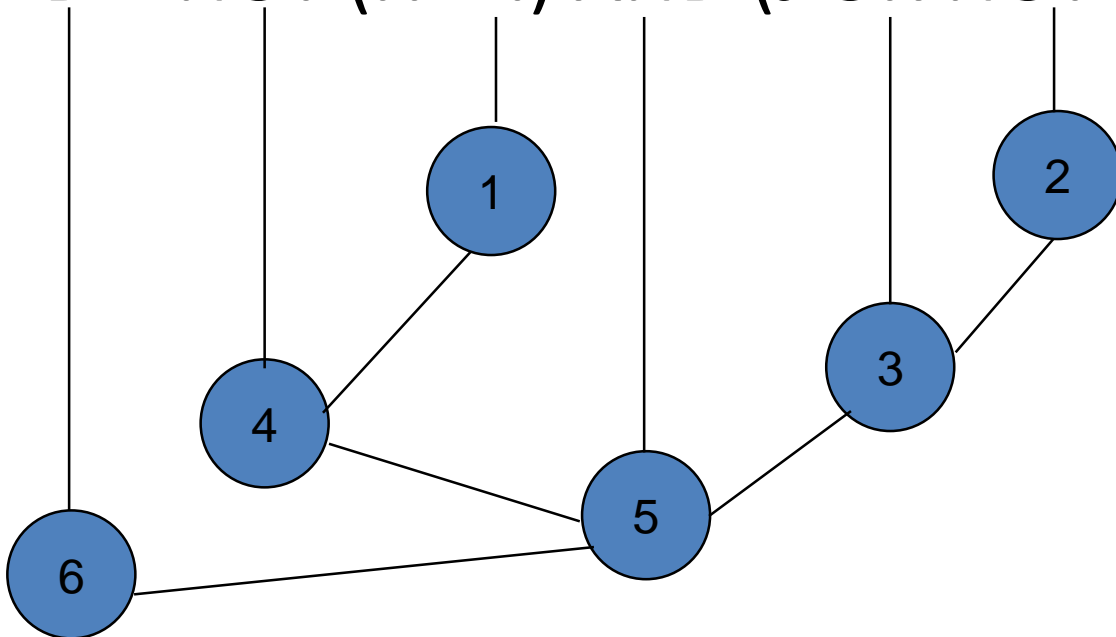
A	B	NOT A	NOT B	NOT A OR B
T	T	F	F	F
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

Example: Equation Uses Both Relational and Logical Operators


Example : $D = \text{NOT} (H < I) \text{ AND } (J \text{ OR } \text{NOT} K)$

assume; $H = 4, I = 2, J = \text{TRUE}, K = \text{FALSE}$

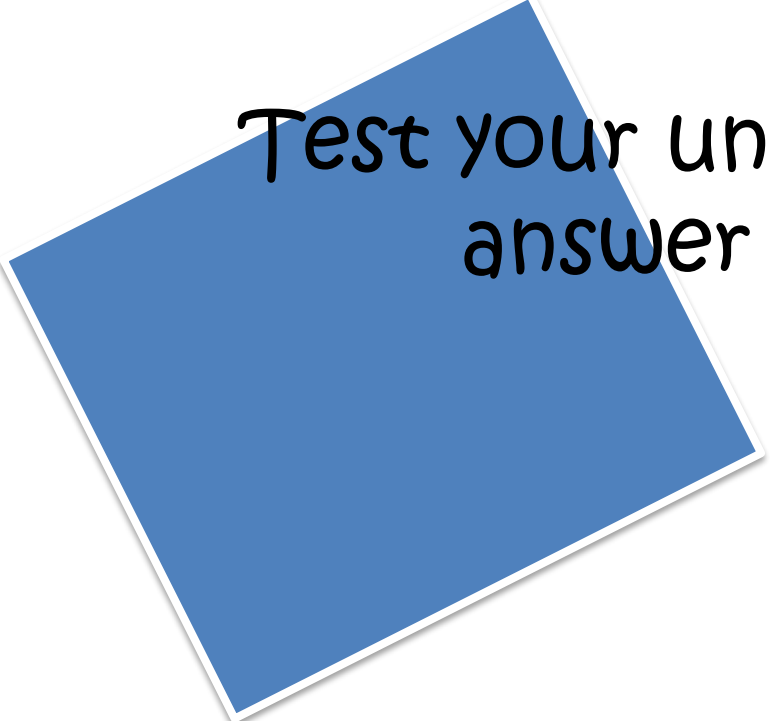
$D = \text{NOT} (H < I) \text{ AND } (J \text{ OR } \text{NOT} K)$



No	Operation	Resultant
1	$H < I$	FALSE
2	NOT K	TRUE
3	J OR resultant of 2	TRUE
4	NOT Resultant of 1 OR Resultant of 2	TRUE
5	Resultant of 4 AND resultant of 3	TRUE
6	Store the resultant of 5 in the memory location called D	



Test your understanding by
answer Tutorial 1



Conclusion / What we have learn today?



Constant VS Variable



Data types (Numeric data: Integer, Real, Character, String, Logical)



Function (Mathematical function, String function etc.)



Operator (Mathematical operators, Relational operators etc.)



Hierarchy of operations



Expressions and Equations

Author Information

NOOR AZIDA BINTI SAHABUDIN

Senior Lecturer

Faculty of Computer Systems & Software Engineering

Universiti Malaysia Pahang

PhD in Educational Technology