

# Algorithm and Data Types

## Contents

Algorithm .....	2
Pseudocode.....	2
Flowchart .....	2
Data Types.....	3
Value Types .....	3
Reference Types .....	4
Basic Terms.....	4

## Algorithm






Susunan langkah untuk menyelesaikan suatu masalah.

## Pseudocode

Versi readable algorithm yang dapat dengan mudah dipahami dan bersifat fleksibel.

## Flowchart

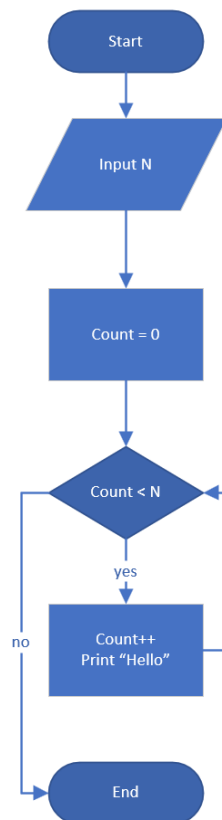
Versi diagram dari algorithm untuk menunjukkan visual flow langkah setiap tahap dalam algorithm.

Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectagle represents a process
	Decision	A diamond indicates a decision

Source: SmartDraw

Misalnya kita mau program yang bilang “Hello” sebanyak jumlah orang yang ada maka algoritmanya bisa seperti berikut.

1. Start
2. Input N (number of people)
3. Set count = 0
4. While 'count' < N:  
    'count' += 1  
    Print "Hello"
5. End



## Data Types

Data types adalah jenis data yang bisa disimpan dalam program, dalam konteks Solidity, yang dimaksud adalah data yang disimpan di smart contract dalam blockchain. Satu istilah yang penting dipahami adalah “**variable**”, variable adalah instance of a data type. Misalnya kita mau menampung unsigned integer yang didefinisikan dengan keyword `uint`, maka kita perlu mendefinisikan variable-nya seperti **`uint myUint`** dimana **`uint`** adalah data type dan **`myUint`** adalah variable name.

## Value Types

Value type adalah tipe data yang langsung disimpan dalam memorynya sendiri dan nilainya akan di-copy jika dilempar ke suatu function atau assignment sehingga nilai atau value original-nya akan selalu terjaga.

- Signed/Unsigned Integer: Data type ini menyimpan angka bulat. Signed integer bisa menyimpan angka negative, sedangkan unsigned integer hanya bisa menyimpan angka positif dari 0.
- Boolean: Data type simpel yang hanya memiliki 2 kemungkinan nilai, **true (1)** atau **false (0)**. Ukurannya hanya 1 byte. Ibarat sebuah **saklar lampu** yang bisa on atau off saja. Boolean bisa digunakan untuk operasi seperti mengecek apakah dua hal itu sama atau berbeda (equality) dan bisa digabung dengan **AND** dan **OR** untuk menggabungkan kondisinya.
- Address: Data type spesial untuk menyimpan account address maupun smart contract address. Ukurannya 20 bytes atau 160 bits. Dalam Solidity, ada **address** biasa dan **address payable**. Address payable digunakan jika perlu mengirim \$ETH ke address tersebut.
- Enum (enumerable): Untuk membuat user-defined type. Nilainya berupa integer constant yang readable dengan constant name. Enum bisa di cast menjadi integer. Ibarat sebuah **label**

untuk hal-hal yang tidak berubah, misal untuk bulan expired snack yang ada di rumah yang dilabeli dengan bulan expired-nya.

- Bytes array: Untuk menyimpan data dalam bentuk byte dengan format **hexadecimal**. Biasa digunakan untuk menyimpan data, seperti integer dan string, yang ukurannya diketahui dan kecil agar menghemat storage. 1 byte atau 8 bits bisa direpresentasikan dengan 2 digit hex. Data type ini cukup jarang ditemukan. Anggap aja bytes itu kayak **rak** yang setiap kotaknya bisa diisi sesuatu, dalam hal ini data.

## Reference Types

Reference type adalah tipe data yang menyimpan lokasi memory dimana datanya disimpan. Berbeda dengan value type yang langsung menyimpan datanya, reference type hanya menyimpan lokasinya.

- String: Untuk menyimpan kumpulan karakter, bisa huruf, angka, atau simbol.
- Struct: User-defined type yang dapat menampung beragam jenis data type. Ibarat suatu lemari di dapur yang didalamnya banyak toples untuk menyimpan beragam jenis bahan dan barang.
- Fixed/Dynamic Array: Menyimpan kumpulan sekelompok data dengan data type yang sama. Fixed berarti jumlah data yang ditampung sudah ditentukan jumlahnya, sedangkan dynamic bisa disesuaikan dengan kebutuhan saat eksekusi.
- Mapping: Menyimpan data dalam bentuk key-value pairs agar mempermudah untuk mencari suatu data atau value berdasarkan key-nya. Anggap ini seperti kamus

## Basic Terms

Algorithm: Kumpulan langkah untuk menyelesaikan masalah atau eksekusi task

Variable: Nama dari instance suatu data type untuk menyimpan data

Data Type: Klasifikasi data yang menunjukkan tipe data yang disimpan variable

Conditional Statement: Statement dalam programming yang mendefinisikan suatu syarat atau kondisi yang biasa digunakan dalam if, else, dan else if

Loop: Control structure yang mengulang satu block of code berulang kali sesuai dengan kondisi yang ditentukan

Function: Block of code yang dapat di-call secara independen untuk menjalankan suatu operasi dan bisa menerima parameter

Parameter: Variable yang dilempar ke suatu function agar datanya dapat digunakan dalam function tersebut

Comment: Catatan yang menjelaskan code untuk dokumentasi. Hal ini diabaikan oleh program

Syntax: Aturan untuk penulisan code dalam programming

Bug: Masalah atau error dalam program

Debugging: Proses untuk mengidentifikasi dan memperbaiki bug

Variable Declaration: Kegiatan mendeklarasi data type dan variable name.