

# DASAR-DASAR PEMROGRAMAN

Materi #02.

Konsep Algoritma dan Pemrograman

Dr. Dede Kurniadi, S.Kom., M.Kom  
Program Studi Teknik Informatika  
Jurusan Ilmu Komputer  
Institut Teknologi Garut



@de2Kurniadi

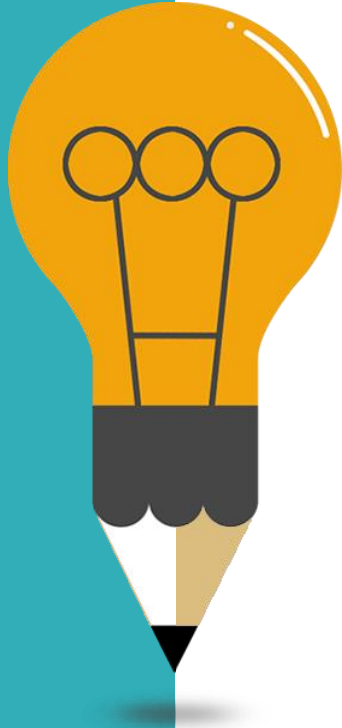


Dede Kurniadi



[www.dedekurniadi.com](http://www.dedekurniadi.com)

# Bahasan Konsep Algoritma dan Pemrograman



**01**

**Algoritma & Pemrograman**

**02**

**Aturan Penulisan Algoritma**

**03**

**Struktur Dasar Algoritma**

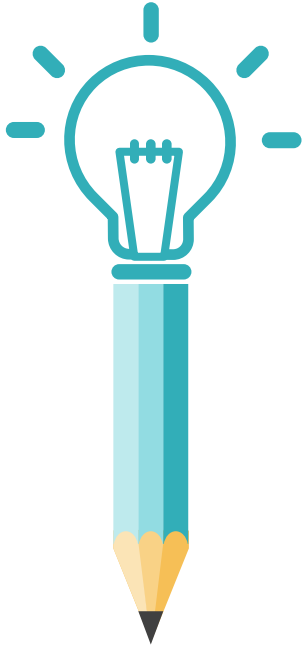
**04**

**Notasi Algoritma**

**05**

**Contoh Kasus**

# ALGORITMA DAN PEMROGRAMAN



Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis dan logis serta dapat ditentukan bernilai salah atau benar.

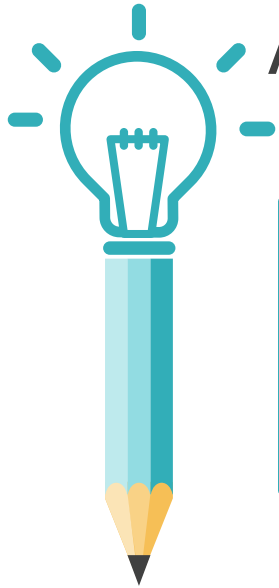
Pemrograman adalah proses menulis, menguji dan memperbaiki (debug), dan memelihara kode yang membangun suatu program komputer (KBBI).



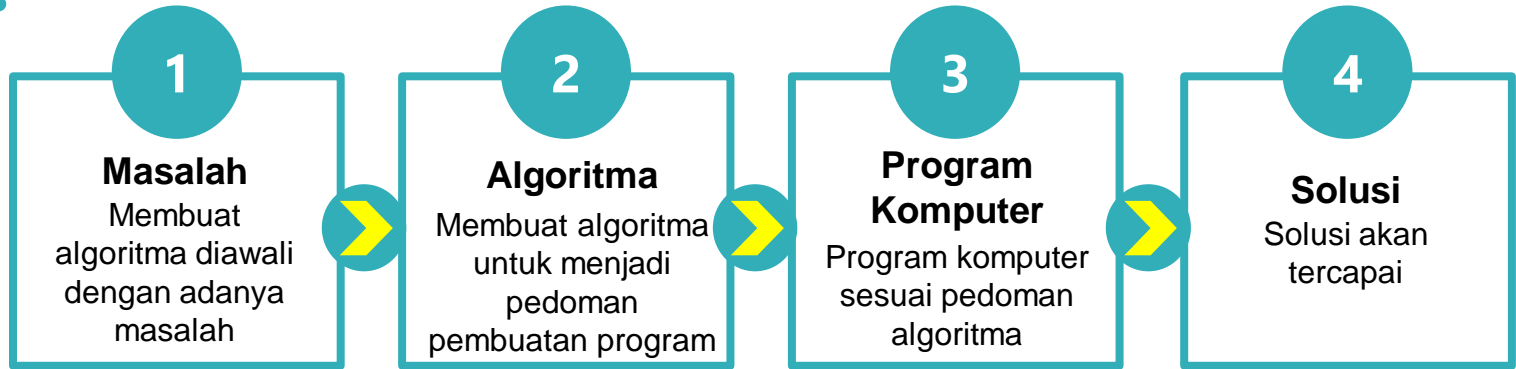
*Muhammad Ibn Musa Al Khawarizmi*

Untuk melakukan pemrograman, diperlukan keterampilan dalam algoritme, logika, bahasa pemrograman, dan pada banyak kasus, pengetahuan-pengetahuan lain seperti matematika sangat diperlukan.

Dengan kata lain Pemrograman adalah suatu seni/ilmu/teknik dalam menggunakan satu atau lebih algoritma yang saling berhubungan dengan menggunakan suatu bahasa pemrograman tertentu sehingga menjadi suatu program komputer. (Dari berbagai sumber)



# ALGORITMA DAN PEMROGRAMAN



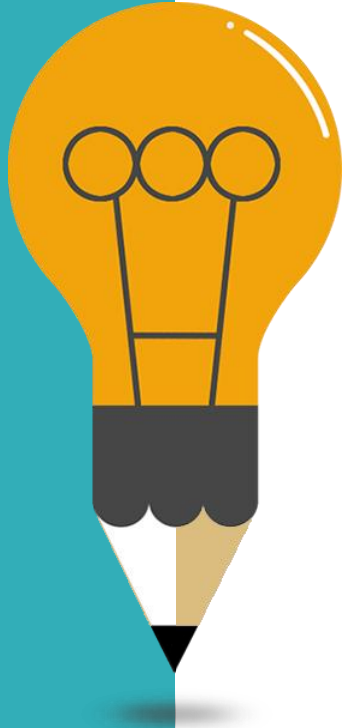
Suatu Algoritma harus dinyatakan dalam bentuk yang dapat dimengerti oleh pemroses/komputer, yaitu:

- Mengerti setiap langkah dalam Algoritma.
- Mengerjakan operasi yang bersesuaian dengan langkah tersebut.

Hal penting dalam membuat sebuah program komputer, yaitu:

- Tujuan pembuatan program.
- Algoritma.
- Bahasa pemrograman itu sendiri.

# Ciri-Ciri Algoritma



1

Algoritma mempunyai awal dan akhir.

2

Setiap langkah harus didefinisikan dengan tepat sehingga tidak memiliki arti ganda.

3

Memiliki masukan (input) atau kondisi awal.

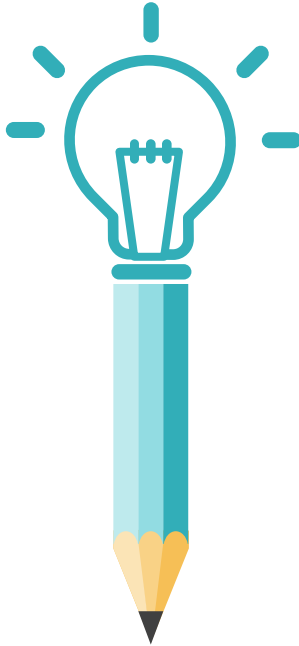
4

Memiliki keluaran (output) atau kondisi akhir.

5

Algoritma harus efektif, bila digunakan benar-benar menyelesaikan persoalan.

# Mekanisme Pelaksanaan Algoritma oleh Pemroses



Komputer hanyalah salah satu pemroses. Agar dapat dilaksanakan oleh komputer, algoritma harus ditulis dalam notasi bahasa pemrograman sehingga dinamakan program.

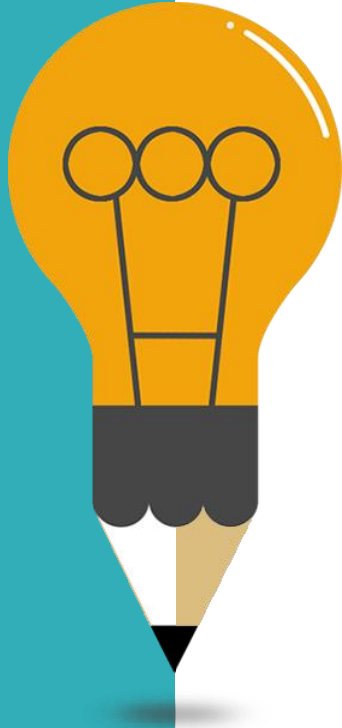
Program adalah perwujudan atau implementasi teknis Algoritma yang ditulis dalam bahasa pemrograman tertentu sehingga dapat dilaksanakan oleh komputer.

Langkah-langkah penyelesaian dalam algoritma dapat ditulis dalam notasi algoritma yang mudah dibaca dan dimengerti.

Agar notasi algoritma mudah ditranslasi ke dalam notasi bahasa pemrograman, maka sebaiknya notasi algoritma tersebut berkoresponden dengan notasi bahasa pemrograman secara umum.

# NOTASI ALGORITMA

Algoritma sebagai langkah-langkah pemecahan masalah dapat dituliskan dengan berbagai cara, yaitu:



**01**

**Uraian Deskriptif**

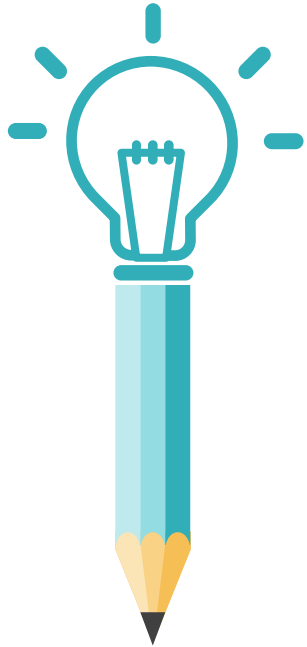
**02**

**Flowchart (Bagan Alir)**

**03**

**Pseudocode**

# Uraian Deskriptif



Uraian Deskriptif adalah Suatu cara menggambarkan masalah dengan diuraikan menggunakan bahasa kita sehari-hari.

Contoh:

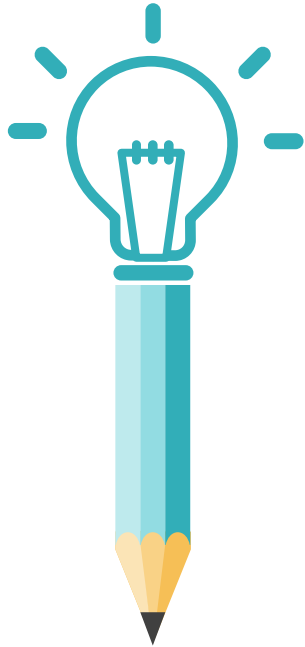
Buatlah algoritma uraian deskriptif untuk menghitung luas Persegi Panjang, Rumus Luas= Panjang x lebar.

Jawaban;

1. Mulai
2. Baca/Input bilangan nilai Panjang dan nilai lebar)
3. Gunakan rumus kalikan nilai panjang dan lebar tersebut
4. Tulis hasilnya
5. Selesai

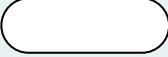






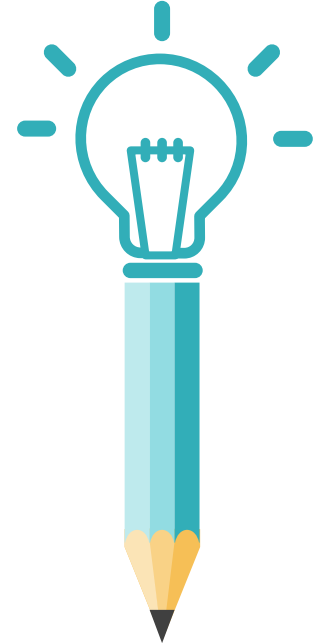
# Flowchart







- Flowchart adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah alur program tersebut.
- Flowchart digunakan dalam pengembangan program aplikasi, merupakan cara tradisional untuk mengajukan tahapan atau langkah-langkah penyelesaian dalam bentuk diagram atau grafis.
- Kelemahan Flowchart
  1. Flowchart hanya cocok untuk masalah yang kecil, untuk masalah yang besar penggunaan flowchart menjadi tidak efisien.
  2. Penggunaan flowchart memerlukan kemampuan menggambar yang baik sehingga penulisan algoritma dengan teknik ini sering kali dalam penyelesaiannya memerlukan waktu yang lama.

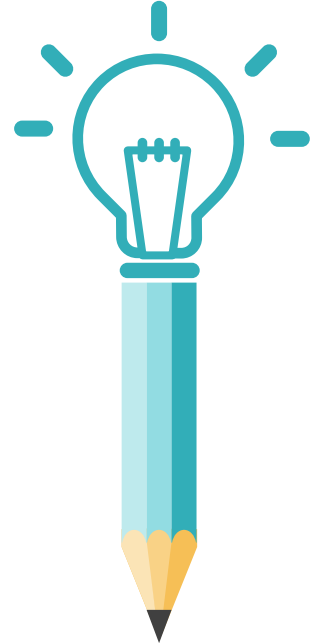
# Simbol-Simbol pada Flowchart

Simbol	Nama	Fungsi
	Terminator	Permulaan/akhir program
	Garis Alir (Flow Line)	Arah aliran program
	Preparation	Proses inisialisasi/pemberian harga awal
	Proses	Proses perhitungan/proses pengolahan data
	Input/Output Data	Proses Input/Output Data, Parameter, Informasi

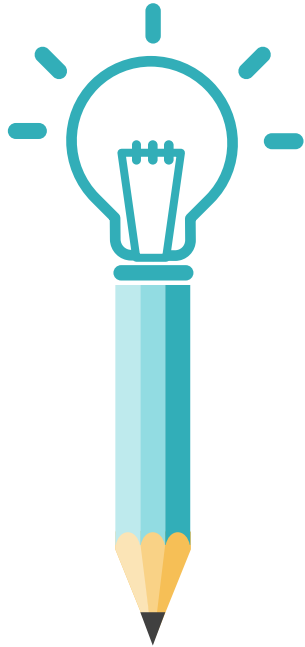


# Simbol-Simbol pada Flowchart

Simbol	Nama	Fungsi
	Predefinied Process (Sub Program)	Permulaan Sub Program/Proses menjalankan sub program
	Decision	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	On Page Conector	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	Off Page Conector	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda



# Flowchart

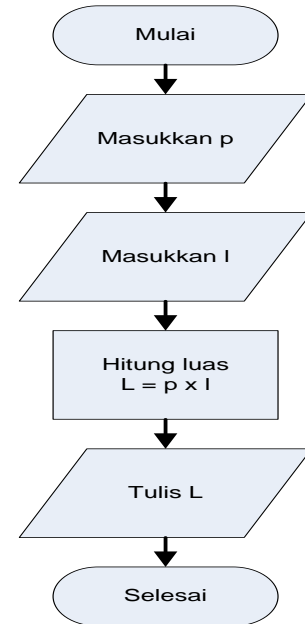


Contoh:

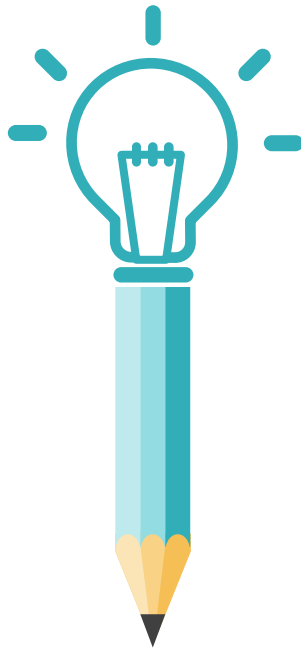
Buatlah algoritma flowchart untuk menghitung luas Persegi Panjang.  
Rumus Luas= Panjang x lebar.

Jawaban;

1. Mulai, simbol Terminator.
2. Masukkan panjang ( $p$ ), *Simbol Input/Output*.
3. Masukkan lebar ( $l$ ), *Simbol Input/Output*.
4. Hitung luas ( $L = p \times l$ ), Simbol Proses
5. Cetak luas ( $L$ ), Simbol Input/Output
6. Selesai, Simbol Terminator



# Pseudo Code



Pseudo Code adalah Kode (pseudo) program atau struktur bahasa yang mendekati notasi suatu bahasa pemrograman.

Contoh:

Buatlah pseudo code untuk menghitung luas Persegi Panjang.  
Rumus Luas= Panjang x lebar.

Jawaban:

Program LuasPersegi

Kamus

panjang,lebar,hasil:integer

Algoritma

Input(panjang,lebar)

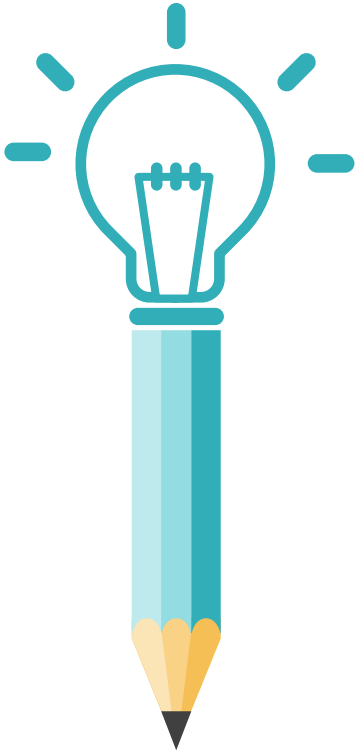
hasil←panjang\*lebar

Output(hasil)



# ATURAN PENULISAN ALGORTIMA

# Aturan Penulisan Algoritma



01

**Judul**  
Header

```
Program Luas_Kubus      ← {Judul Algoritma}  
{ Menghitung luas kubus untuk ukuran sisi yang dibaca dari piranti masukan lalu  
  mencetak hasilnya kepiranti keluaran }      ← {Spesifikasi Algoritma}
```

02

**Kamus**  
Deklarasi

## Kamus

{ Pada bagian ini, didefinisikan nama konstanta, nama variable, nama prosedur dan nama fungsi }

03

**Algoritma**  
Deskripsi

## Algoritma

{ Pada bagian ini algoritma dituliskan. Semua teks yang dituliskan tidak diantara tanda kurung kurawa akan dianggap sebagai notasi algoritma yang akan berpengaruh terhadap kebenaran algoritma }

## Komentar

Pada setiap bagian tersebut apabila akan dituliskan komentar mengenai setiap bagian tersebut dituliskan diantara tanda kurung kurawal {Komen}.

Notasi algoritma yang dituliskan diantara tanda ini tidak akan dieksekusi oleh program.

# Contoh Penyelesaian Masalah dengan Algoritma:



Mencetak String “Selamat Belajar Algoritma dan Pemrograman” ke piranti Keluaran.

**Program** Cetak\_string

{mencetak string “Selamat Belajar Algoritma dan Pemrograman” ke piranti  
keluaran}

**Kamus**

{tidak ada}

**Algoritma**

**Output** ('Selamat Belajar Algoritma dan Pemrograman')



# Contoh Penyelesaian Masalah dengan Algoritma:

Menentukan nilai terbesar dari bilangan bulat yang dibaca dari piranti masukan dan menuliskan hasilnya ke piranti keluaran.



## Program Nilai\_Maksimal

{Menentukan nilai tertinggi yang dibaca dari piranti masukan dan hasilnya dicetak ke piranti keluaran}

## Kamus

hasil,x,y : **integer** {hasil merupakan variabel untuk menampung nilai keluaran}  
{x,y adalah variabel untuk menampung nilai masukan}

## Algoritma

**input** (x,y) {membaca nilai x dan y dari piranti masukan}

**if** x < y **then** {operasi kondisional}

    hasil ← x {hasil di *assignment* oleh nilai terbesar}

**else**

    hasil ← y

**output** (hasil) {nilai didalam variabel hasil dicetak ke piranti keluaran}

# Struktur Dasar Algoritma

01

**Struktur Runtunan  
(Sequence)**

- Struktur Runtunan : digunakan untuk program yang pernyataannya sequential atau urutan.

02

**Struktur Pemilihan  
(Selection)**

- Struktur Pemilihan : digunakan untuk program yang menggunakan pemilihan atau penyeleksian kondisi.

03

**Struktur Perulangan  
(Iteration)**

- Struktur Perulangan : digunakan untuk program yang pernyataannya dieksekusi berulang-ulang.

# Latihan

## Contoh 1 : Kasus dan Penyelesaian Algoritma



**Menghitung luas dan keliling lingkaran. Proses kerjanya sebagai berikut: (Algoritma Uraian Deskripsi)**

**Jawaban:**

1. Baca jari-jari lingkaran
2. Tentukan konstanta  $\phi = 3.14$
3. Hitung luas dan keliling
  - $L = \phi * r * r$
  - $K = 2 * \phi * r$
4. Cetak luas dan keliling

**Algoritma konversi suhu dalam derajat Celcius ke derajat Kelvin  
Penyelesaian menggunakan Pseudo Code:**

**Jawaban:**

Input (Celcius)

$Kelvin = Celcius + 273$

Output (Kelvin)

# Latihan

## Contoh 2 : Kasus dan Penyelesaian Algoritma



**Menghitung rata-rata tiga buah data, menggunakan algoritma dalam struktur bahasa Indonesia dan Pseudo Code**

### **Algoritma dengan struktur bahasa Indonesia**

Baca bilangan a, b, dan c

Jumlahkan ketiga bilangan tersebut

Bagi jumlah tersebut dengan 3

Tulis hasilnya

### **Algoritma dengan Pseudo Code**

input (a, b, c)

Jml =  $a+b+c$

Rata =  $Jml/3$

Output (Rata)

# Latihan

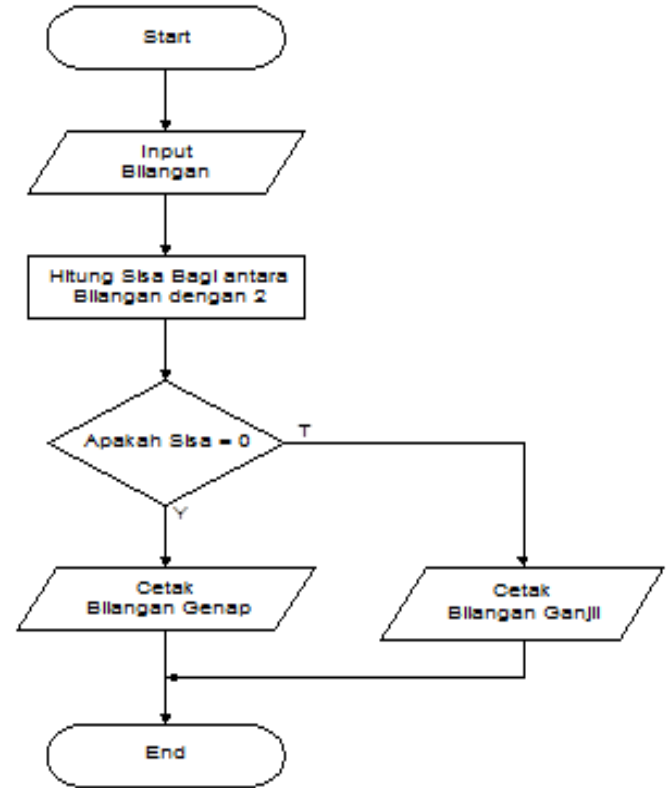
## Contoh 3 : Kasus dan Penyelesaian Algoritma

Buatlah algoritma dan flowchart untuk menentukan apakah suatu bilangan merupakan bilangan ganjil atau bilangan genap.

**Algoritma dalam uraian deskriptif :**

1. Tentukan bilangan yang akan diinputkan.
2. Hitung sisa bagi antara bilangan dengan 2.
3. Bila sisa hasil bagi sama dengan 0, maka bilangan itu adalah bilangan genap tetapi bila sisa hasil bagi sama dengan 1, maka bilangan itu adalah bilangan ganjil.

**Penyelesaian dengan menggunakan Flowchart, dapat dilihat pada gambar disamping**



# Latihan

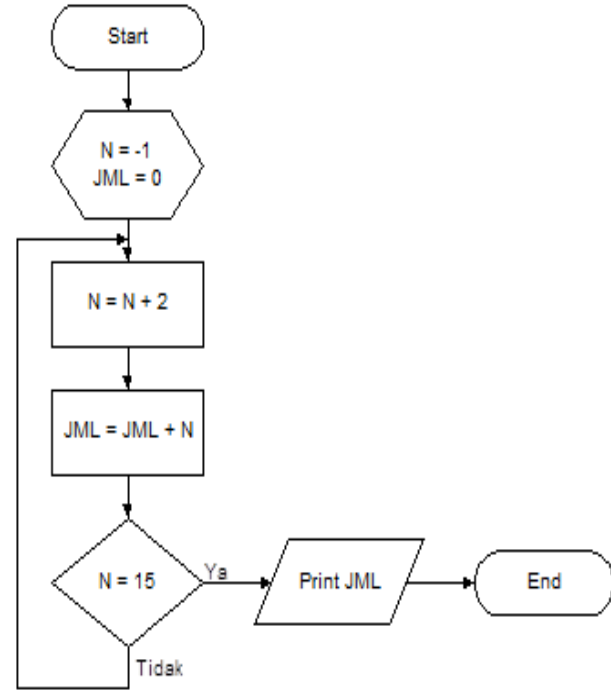
## Contoh 4 : Kasus dan Penyelesaian Algoritma

Berikut ini contoh algoritma dan flowchart untuk mencari hasil perhitungan  $1+3+5+7+9+11+13+15$ .

### Langkah-Langkah Penyelesaian (Algoritma)

1. Tentukan nilai awal untuk  $N = -1$  dan  $JML = 0$ .
2. Tentukan nilai beda bilangan  $N$ . ( $N = N + 2$ )
3. Hitung hasil perhitungan jumlah bilangan. ( $JML = JML + N$ )
4. Bandingkan nilai  $N$ , apakah nilai  $N = 15$ .
5. Jika  $N = 15$ , berarti perhitungan jumlah bilangan sudah terpenuhi, goto 7.
6. Jika  $N \neq 15$ , berarti perhitungan jumlah bilangan belum terpenuhi, goto 2.
7. Cetak nilai  $JML$ , yaitu hasil perhitungan jumlah bilangan  $1 + 3 + 5 + 7 + 9 + 11 + 13 + 15$ .

Penyelesaian dengan menggunakan Flowchart, dapat dilihat pada gambar disamping





Sekian dan Terima Kasih