

# Pemrograman Visual dengan Flowgorithm

Tim Pengajar PKS 2

Institut Teknologi Sumatera

# Outline

- Pemrograman Visual
- Pemrograman Visual dengan Flowgorithm
- Latihan

# Pemrograman Visual

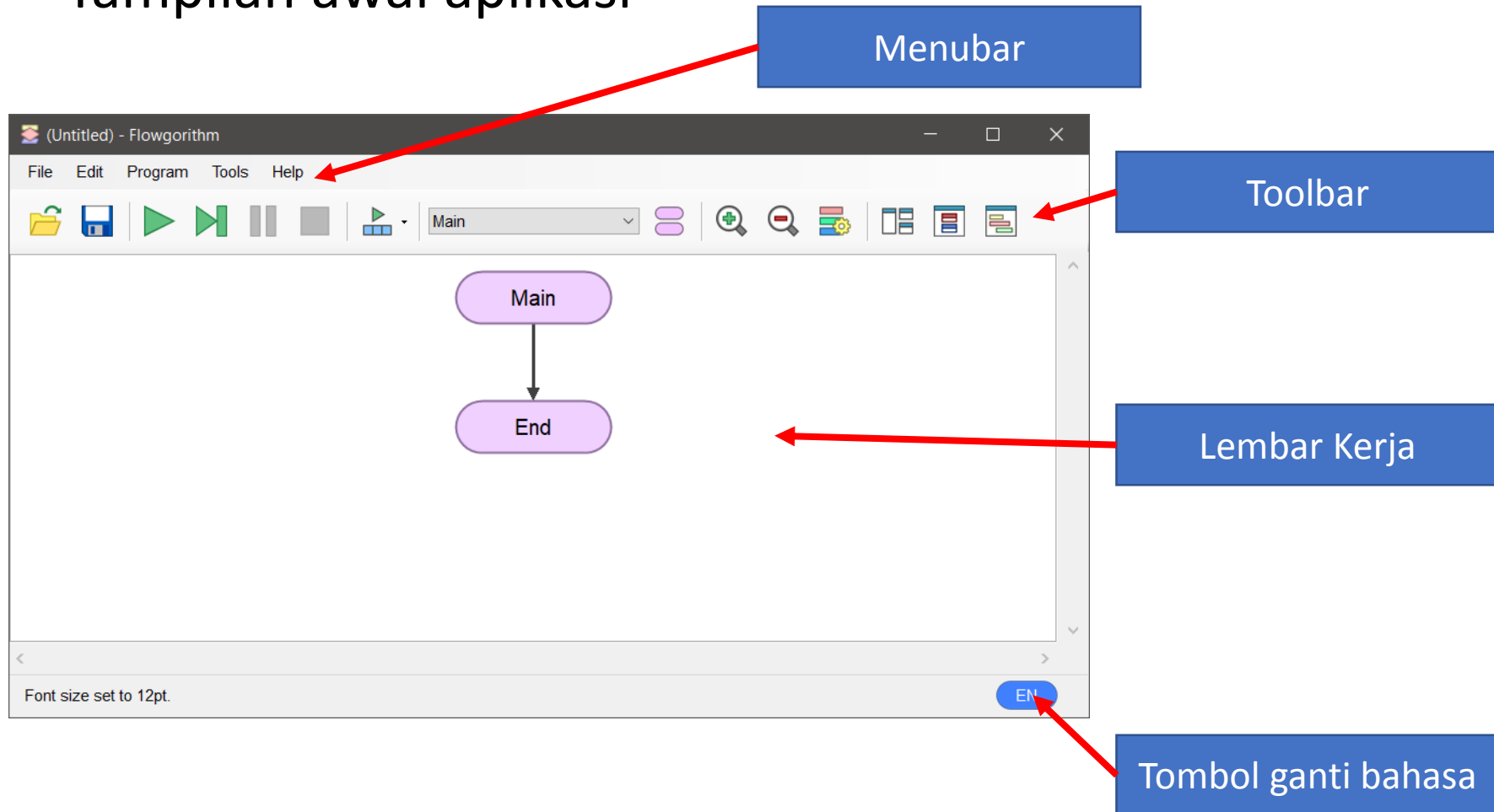
- Program biasanya dibuat dengan menggunakan bahasa pemrograman, seperti Pascal, C/C++, Java, Python, dll.
- Setiap bahasa pemrograman memiliki struktur sintaks/kosa kata yang terkesan sangat tidak ramah bagi *programmer* pemula.
- Pemrograman visual memungkinkan *programmer* memanipulasi elemen program secara visual sehingga lebih intuitif bagi pemula.
- Contoh aplikasi pemrograman visual ***Scratch***, ***App Inventor***, dan ***Flowgorithm***.

# Pengenalan Flowgorithm

- **Flowgorithm** merupakan aplikasi pembelajaran pemrograman bagi pemula.
- Aplikasi ini Gratis!
- Pemrograman dilakukan secara visual dengan memanfaatkan bagan *flowchart* sederhana.
- Kalian tentu sudah paham cara membuat *flowchart* pada pertemuan sebelumnya kan?! 😊

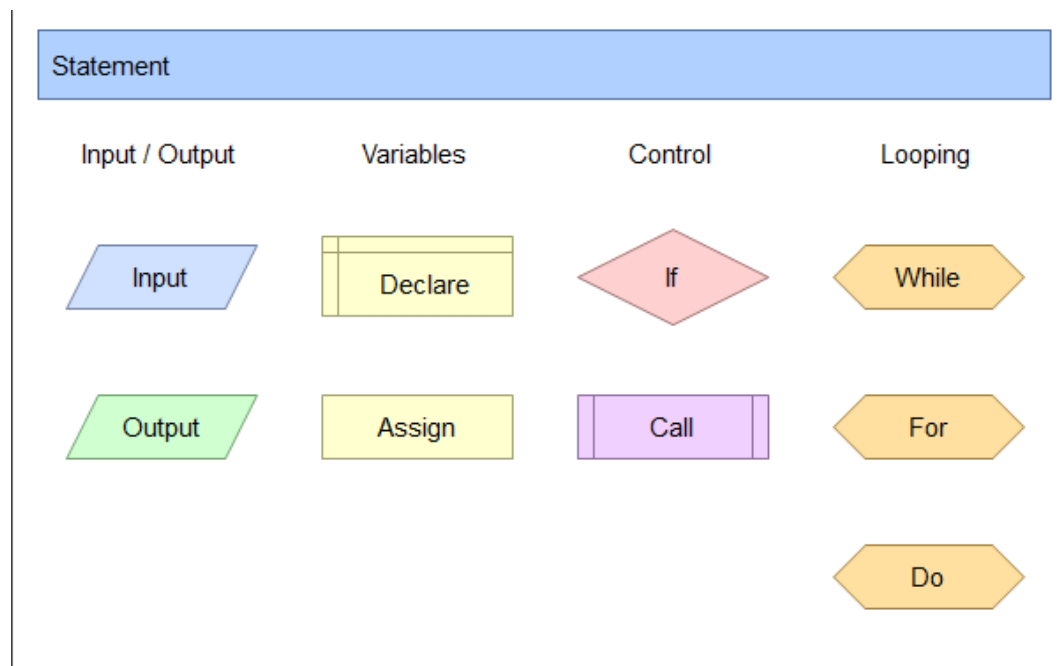
# Flowgorithm

- Tampilan awal aplikasi



# Komponen Aplikasi

- Elemen dalam Flowgorithm hampir sama dengan elemen pada flowchart biasa!

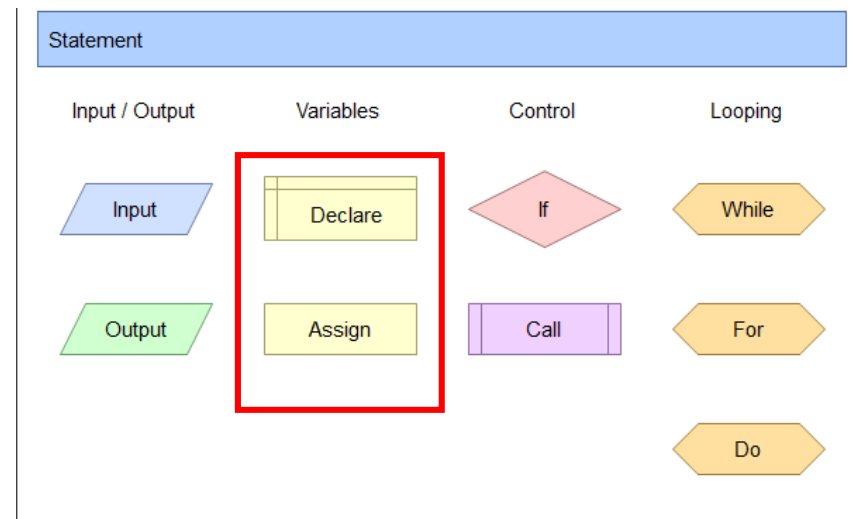


# Variabel

- Dalam *flowchart* biasa kita tidak diperkenalkan dengan konsep variabel.
- Untuk menyimpan data, digunakan variabel, dimana variabel memiliki tipe data.
- Namun jangan khawatir! Dalam perkuliahan kali ini, kita hanya menggunakan tipe data bilangan bulat (Integer) dan bilangan Real. Tentu kalian semua sudah paham.

# Variabel (2)

- Dalam pemrograman, **Variabel** harus dideklarasikan terlebih dahulu sebelum digunakan.
- Untuk deklarasi variabel pada **Flowgorithm**, gunakan elemen **Declare**.
- Untuk memberikan nilai pada variabel, gunakan elemen **Assign**.





# Operasi

- Serupa dengan Microsoft Excel, untuk memproses data, digunakan operasi **aritmatika** dan **logika**.
- Gambar disamping merupakan operasi pada **Flowgorithm** dengan prioritas evaluasinya dari yang tertinggi hingga terendah.

Level	Name	Operators
8	Unary	- ! not
7	Exponent	^
6	Multiply	* / % mod
5	Addition	+ -
4	Concatenate	&
3	Relational	> >= < <= == = != <>
2	Logical And	and &&
1	Logical Or	or

# Dokumentasi Flowgorithm

- Dalam Flowgorithm juga terdapat beberapa konstanta dan fungsi *built-in* yang dapat dimanfaatkan dalam pemrosesan data lebih lanjut.
- Dokumentasi dan Tutorial dapat diakses pada laman berikut:

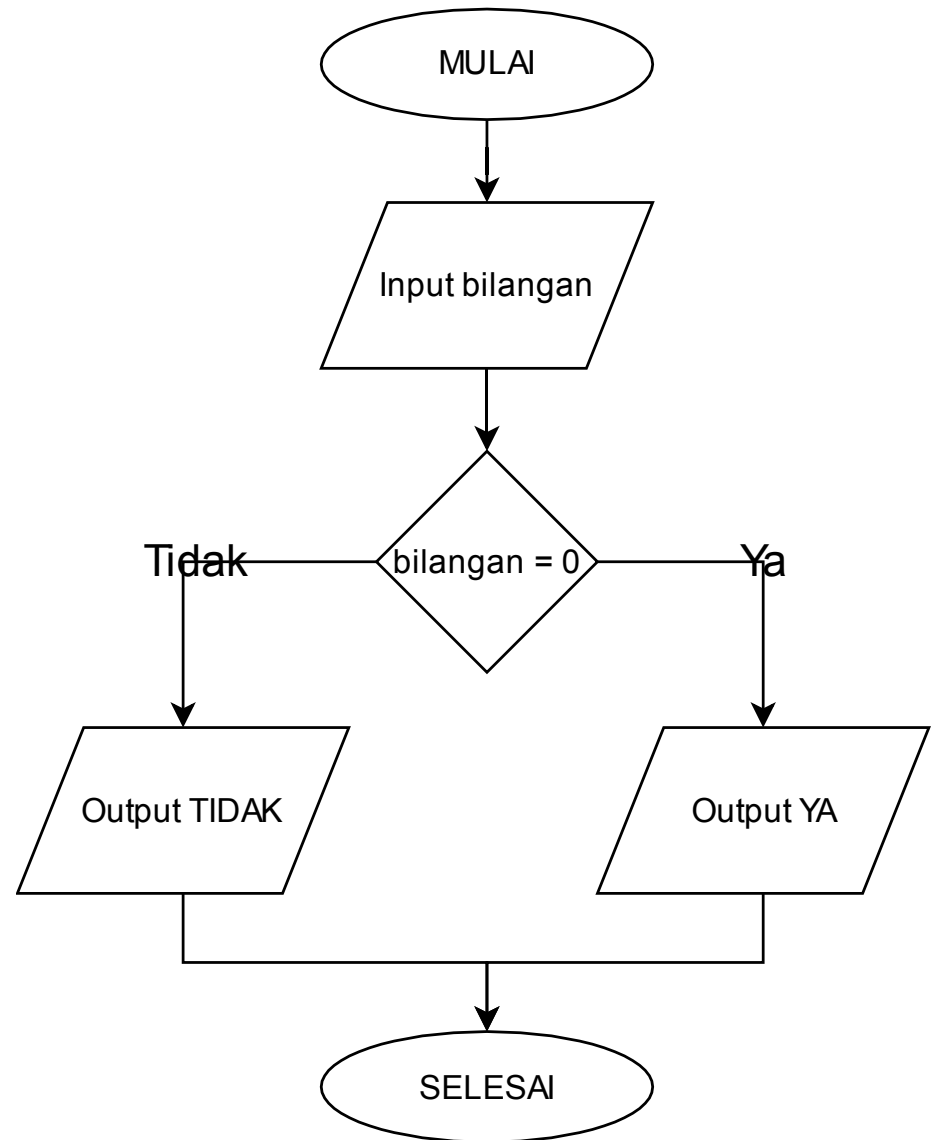
<http://flowgorithm.org/documentation/index.htm>

# Latihan

- Kita akan membuat flowchart untuk mengecek apakah bilangan bulat yang dimasukan oleh pengguna adalah bilangan 0.
- Bagaimanakah bentuk flowchart-nya?

# Langkah 1

- Rancang bentuk *flowchart* secara umum



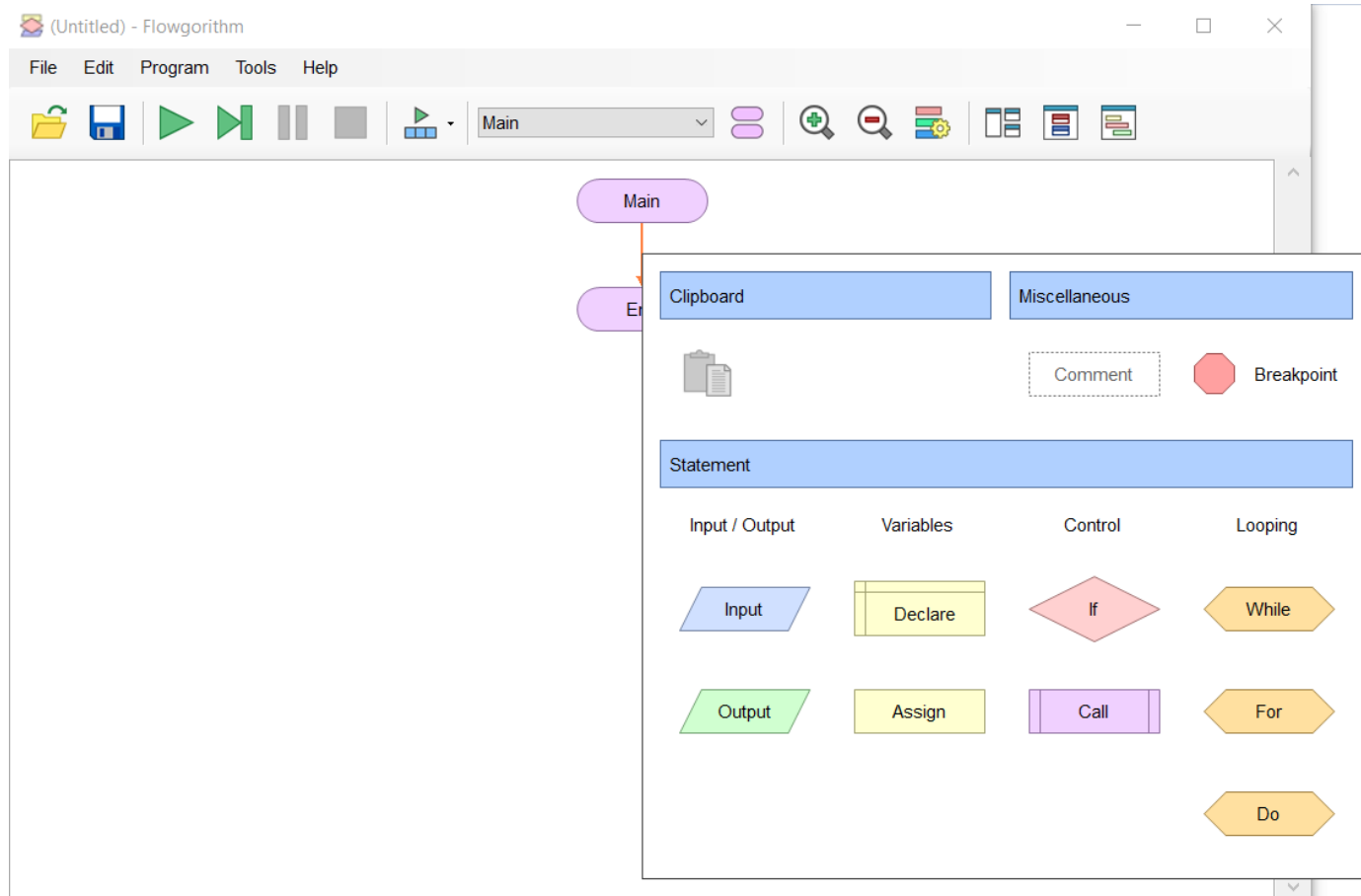
# Langkah 2

- Buka aplikasi **Flowgorithm**
- Lalu muncul tampilan halaman awal aplikasi seperti yang ditampilkan pada slide sebelumnya.

# Langkah 3

- Sorot panah antara blok **START** dan **END** sehingga warnanya berubah menjadi oranye.
- Kemudian, klik tanda panah tersebut sehingga muncul menu komponen

# Langkah 3 (Ilustrasi)

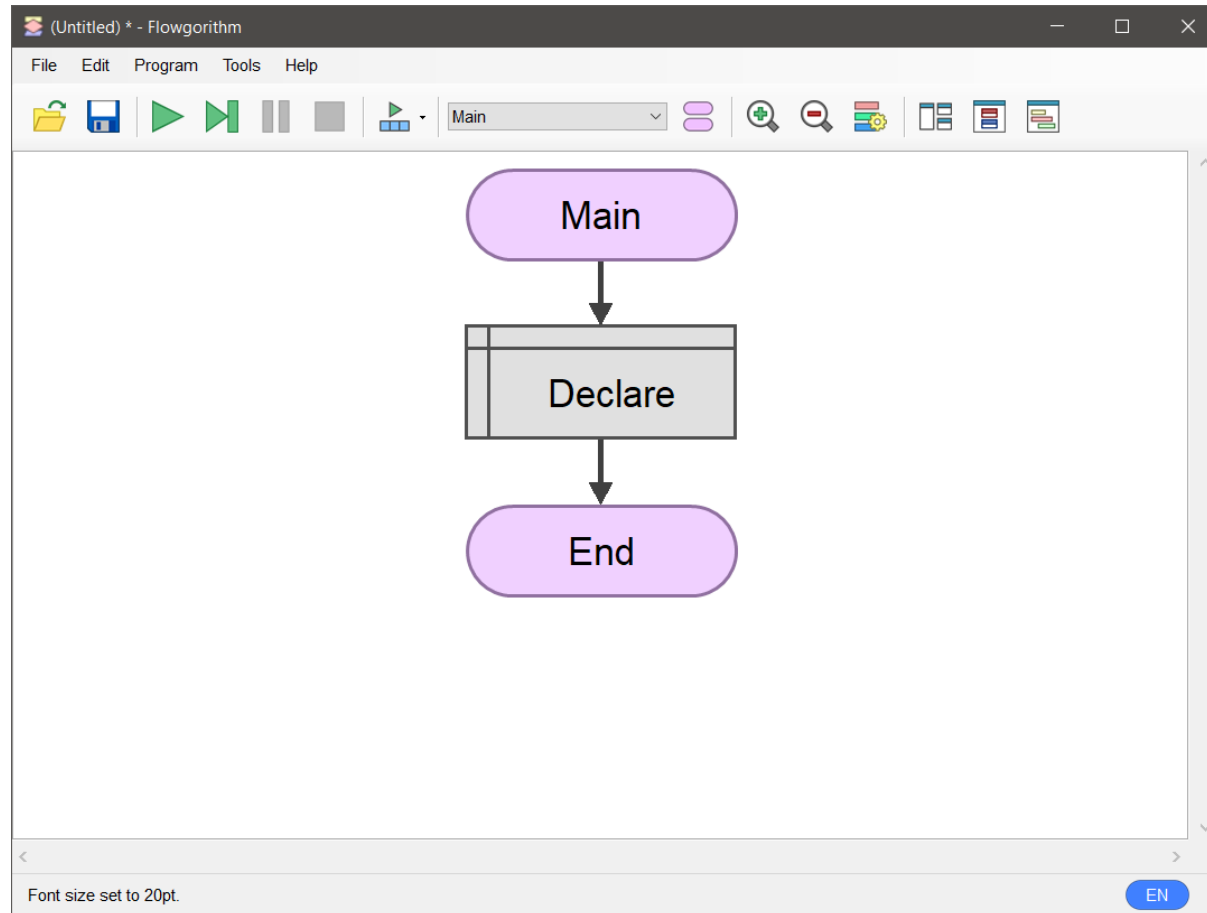


# Langkah 4

- Sesuai spesifikasi, kita akan meminta input dari pengguna dan menyimpan data tersebut untuk dicek.
- Data disimpan di variabel bilangan.
- Jangan lupa deklarasi variabel terlebih dahulu sebelum digunakan!
- Klik elemen **Declare** pada menu komponen.



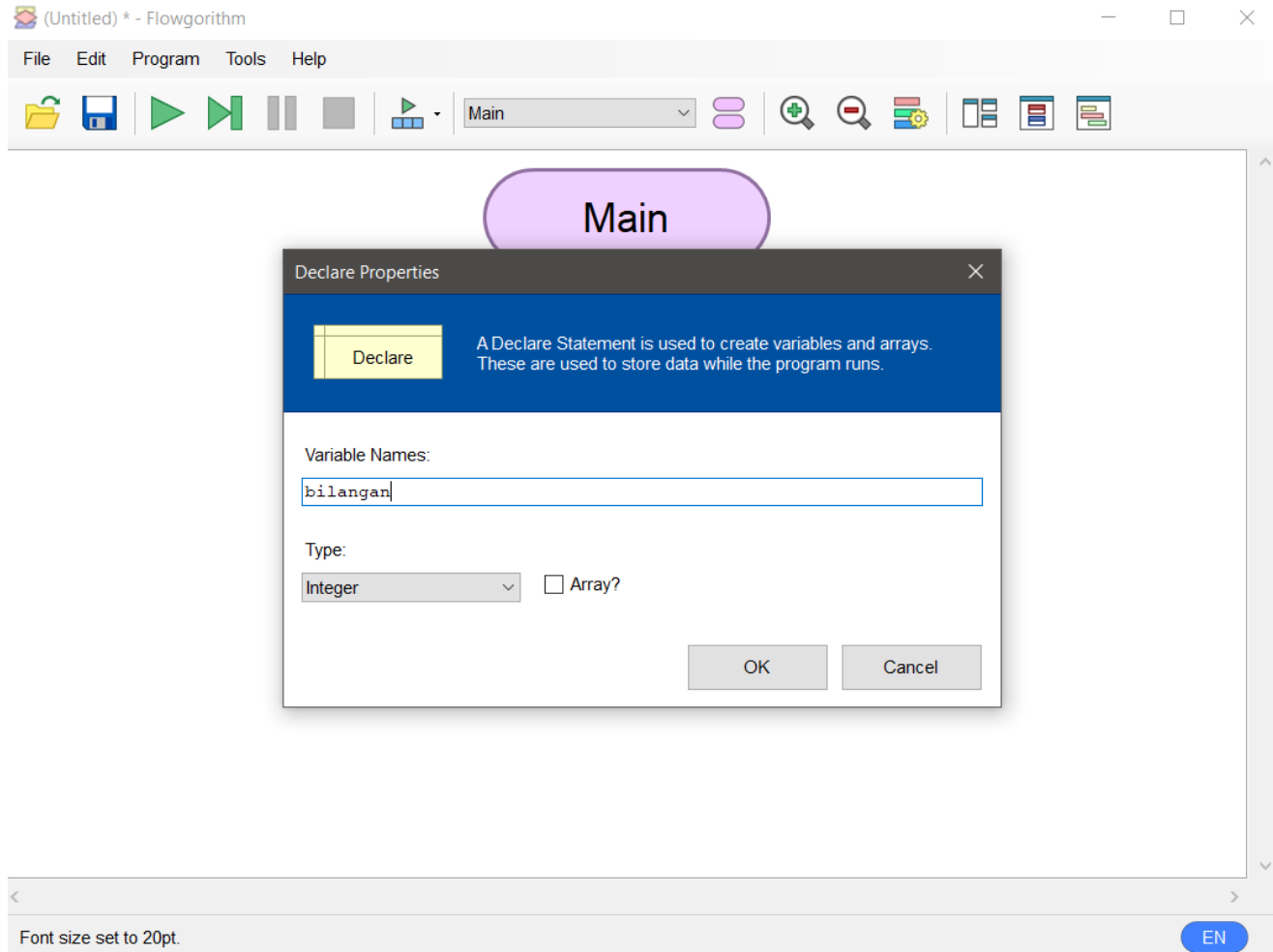
# Langkah 4 (Ilustrasi)



# Langkah 5

- Setelah itu, Anda harus memberikan nama variabel untuk membedakannya dengan variabel lainnya dalam program.
- Klik dua kali pada elemen **Declare**, lalu berikan nama variabel **bilangan**.
- Pilih tipe bilangan yang sesuai. Kali ini kita akan mengecek bilangan bulat atau Integer.
- Setelah selesai, klik OK.
- Perhatikan, elemen **Declare** berubah menjadi **Integer bilangan**.

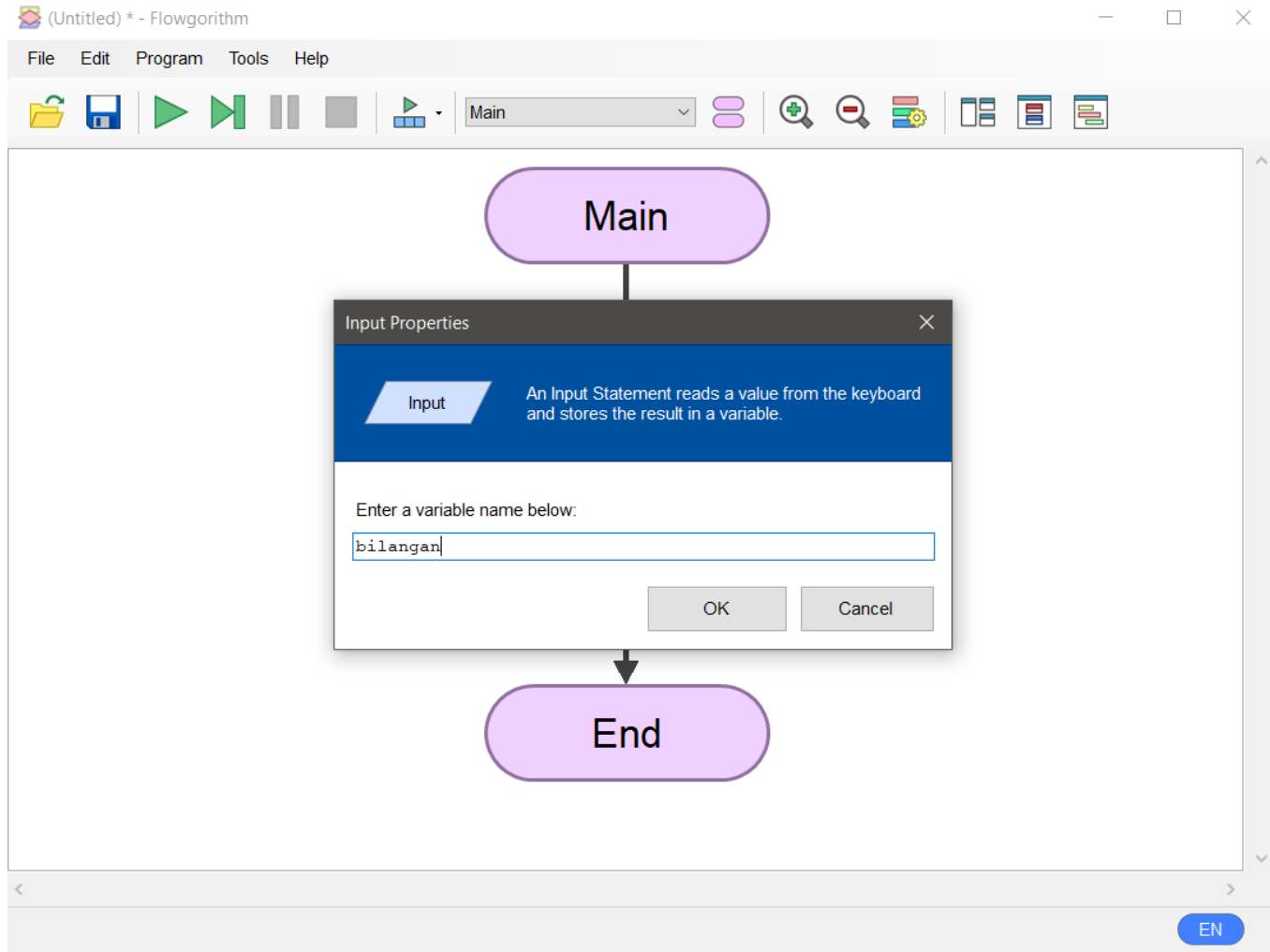
# Langkah 5 (Ilustrasi)



# Langkah 6

- Variabel sudah dideklarasikan, sekarang kita dapat menerima input dari pengguna.
- Klik tanda panah antara deklarasi variabel dengan **END** untuk menambahkan elemen **Input**.
- Bilangan masukan dari pengguna akan disimpan ke variabel **bilangan**, sehingga kita harus memberikan nama variabel bilangan pada elemen **Input**.
- Caranya, klik dua kali pada elemen Input, lalu ketik **bilangan** (nama variabel).
- Setelah selesai, klik OK

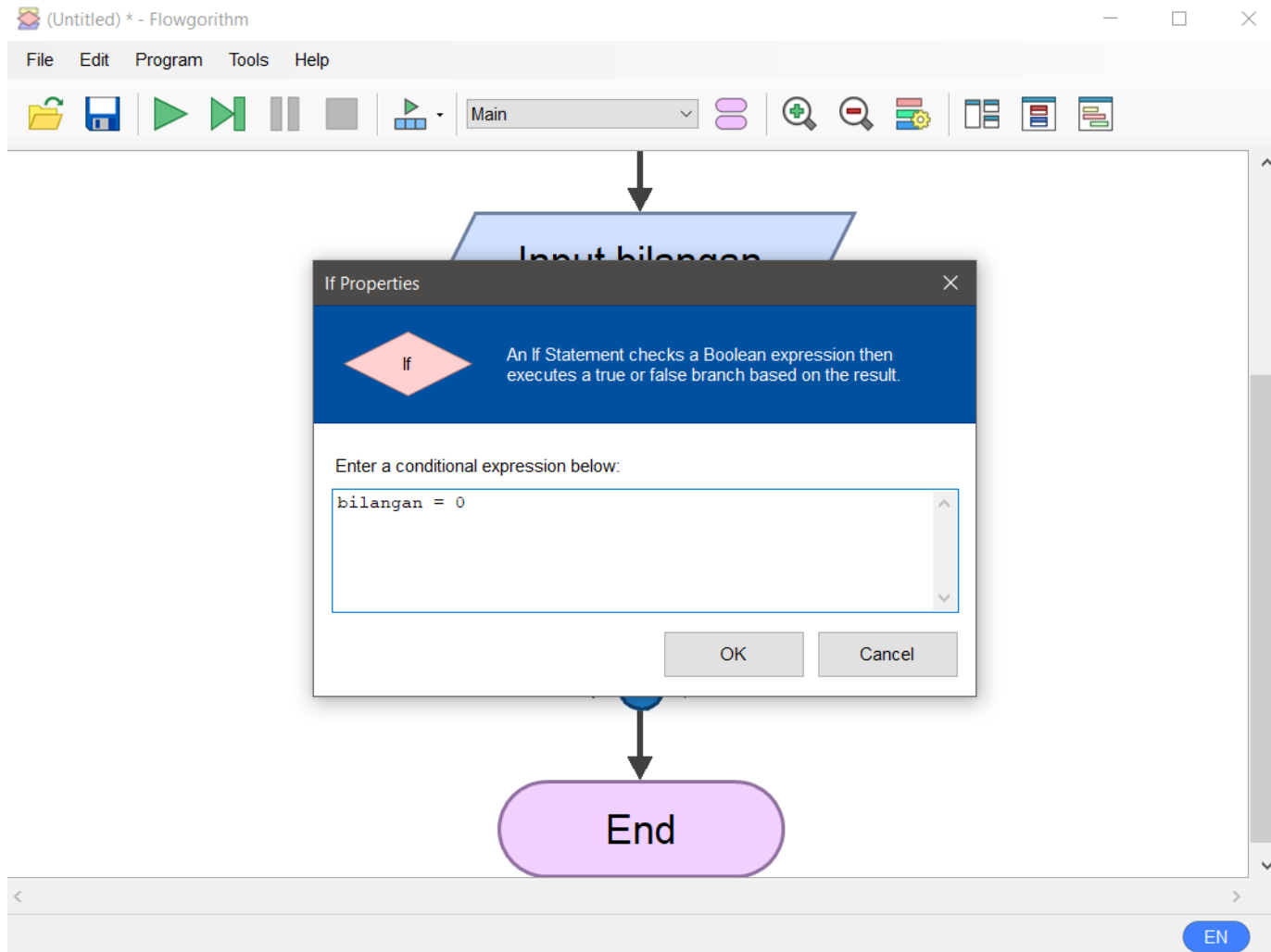
# Langkah 6 (Ilustrasi)



# Langkah 7

- Selanjutnya, kita akan mengecek apakah bilangan sama dengan 0.
- Pada tanda panah antara input dan **END**, tambahkan elemen kontrol **IF**.
- Berikan kondisi pengecekan pada elemen IF dengan cara klik dua kali pada elemen.
- Kondisi yang dicek adalah apakah bilangan input sama dengan 0 atau **bilangan = 0**.
- Lalu klik OK.

# Langkah 7 (Ilustrasi)

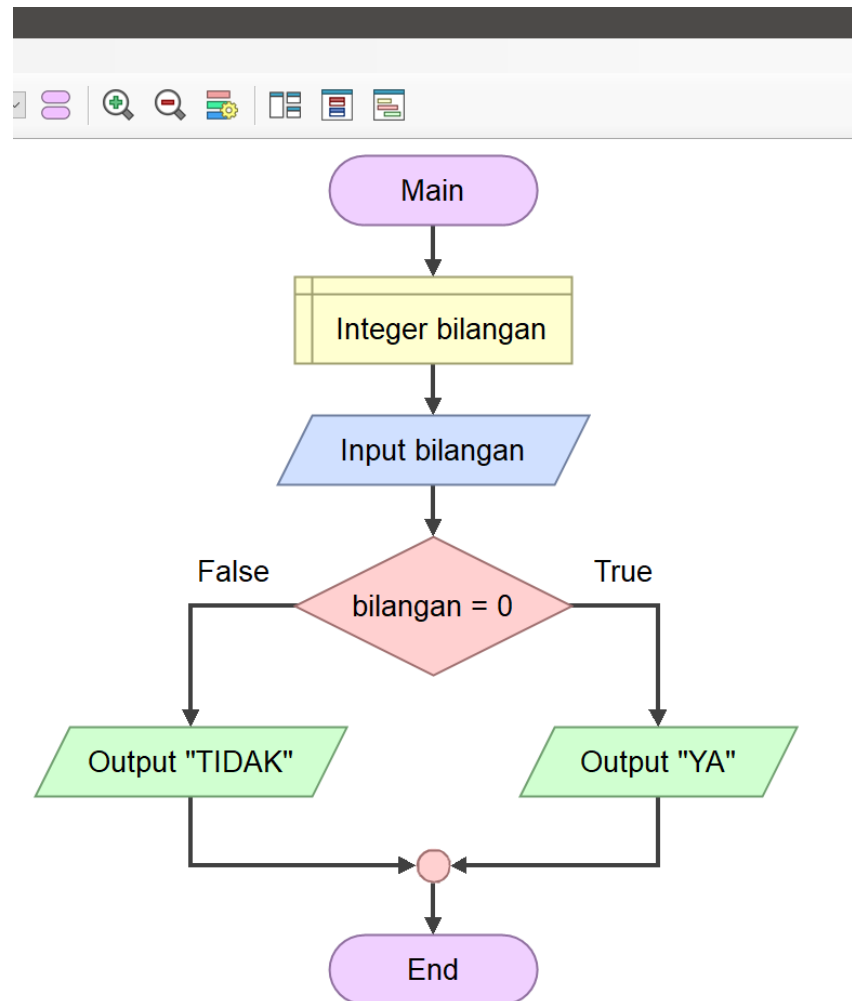


# Langkah 8

- Setelah bilangan dicek, kita akan menentukan apakah bilangan input adalah 0 atau bukan.
- Pada arah panah **TRUE**, yang artinya kondisi **bilangan = 0** bernilai **benar**, kita akan memberikan elemen **Output**, yang berisi teks **“YA”**.
- Pada arah panah **FALSE**, yang artinya kondisi **bilangan = 0** bernilai **salah**, kita akan memberikan elemen **Output**, yang berisi teks **“TIDAK”**.
- Caranya sama dengan menambah komponen lainnya. Silakan Anda lakukan secara mandiri.
- ~**Perhatikan**: teks harus diapit tanda petik dua



# Tampilan program utuh



# Menghapus elemen

- Untuk menghapus elemen, caranya dengan klik kanan pada elemen yang akan dihapus.
- Klik menu **Delete**.
- Komponen akan terhapus dari program.

# Eksekusi program

- Untuk melihat apakah program berjalan sesuai dengan spesifikasi, **Flowgorithm** memungkinkan programmer untuk melakukan eksekusi program dalam aplikasi.
- Caranya, klik tombol **Run** pada toolbar atau tekan tombol **F5** pada keyboard.
- Jika sukses, selanjutnya akan muncul jendela **Console**.
- Berikan masukan bilangan **0**, lalu tekan enter. Jika program benar, **Console** akan menampilkan teks **YA**.

# Contoh tampilan eksekusi

YA

0

Enter

# Tugas

- Buatlah sebuah program yang dapat menerima masukan **bilangan bulat** dan kemudian menentukan apakah bilangan tersebut **habis dibagi 3 DAN tidak habis dibagi 2!**
- **Tantangan**: Tambahkan sebuah mekanisme dimana saat program hendak berakhir, program menanyakan kepada pengguna apakah pengguna ingin memasukkan sebuah bilangan lagi. Jika pilihan **YA**, maka program akan kembali ke awal yaitu meminta input pengguna.

# AKHIR SLIDE

Terimakasih!

Selamat Belajar!