

การเขียนโปรแกรมโดยใช้ภาษาสั้นๆ ลักษณะ

# flowchart basics

## ขั้นตอนการเขียนโปรแกรม

1. ออกรูปแบบขั้นตอนการทำงานโปรแกรม โดยใช้ **flowchart** หรือ **pseudo code**
2. แปลงจาก **flowchart** หรือ **pseudo code** เป็น **source code** ภาษาโปรแกรม
3. ทดสอบโปรแกรม

# Flowchart

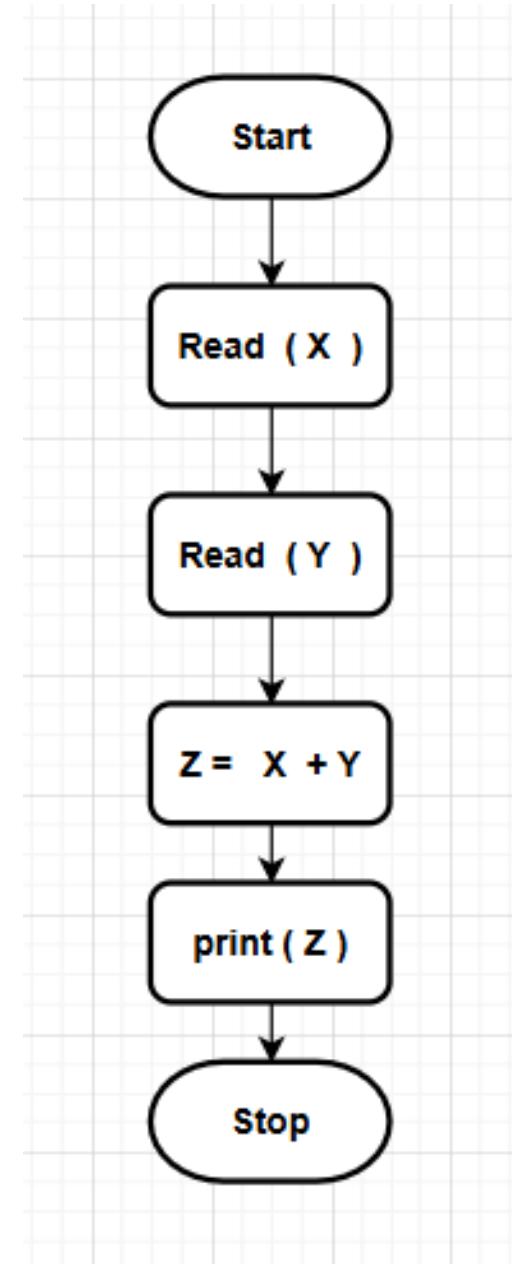
- เครื่องมือในการบรรยายการทำงานของ statement ของโปรแกรม

- Symbol

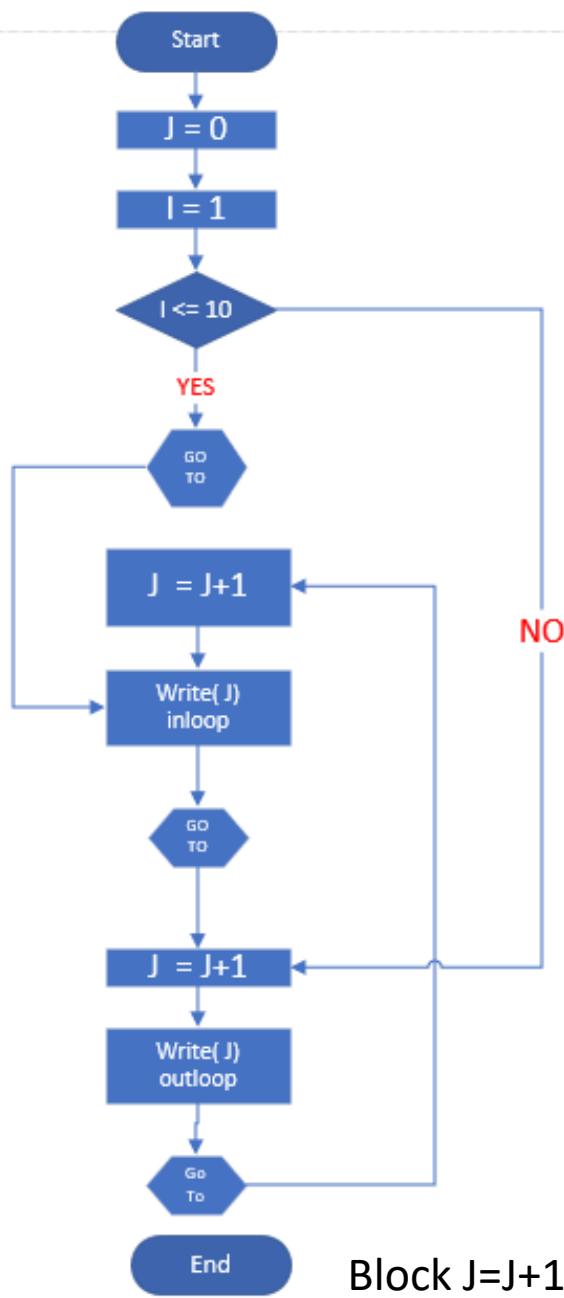
Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Process	A rectangle represents a process
	Decision	A diamond indicates a decision

# หลักการ execution flow

- เริ่มจากสัญลักษณ์ start/begin
- Execution ตามสัญลักษณ์ลูกศร
- หยุดการเมื่อพบสัญลักษณ์ stop/end



## PROBLEM NONSTRUCTURE FLOW



Problem : Infinite loop

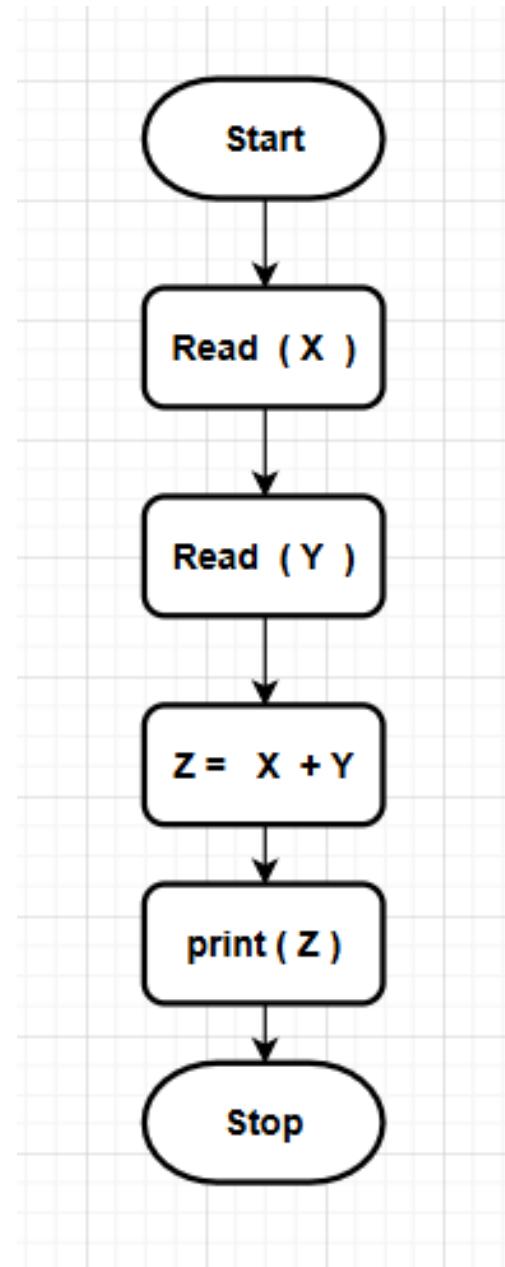
Block  $J=J+1$  mainly flows from  $J=J+1$

# Structure Programming Flows

- Sequential Structure : ทำทีละ步骤ๆ จากบนลงล่าง
- Selection / Decision Structure : ทำตามเงื่อนไข
- Loop / Repetition Structure : ทำซ้ำหลายครั้ง

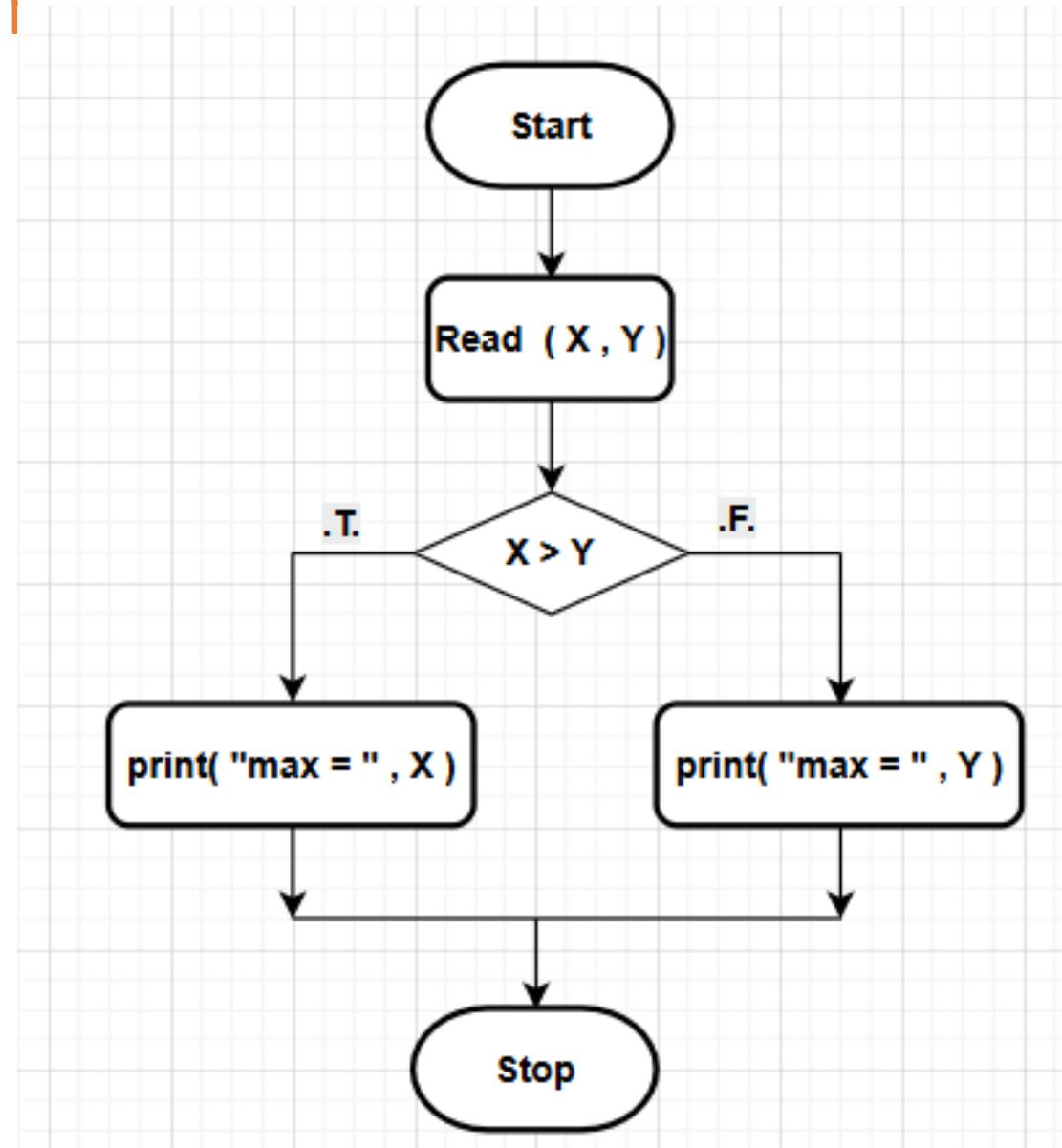
# Sequential Structure

ทำทีละ步รหัด จากบนลงล่าง



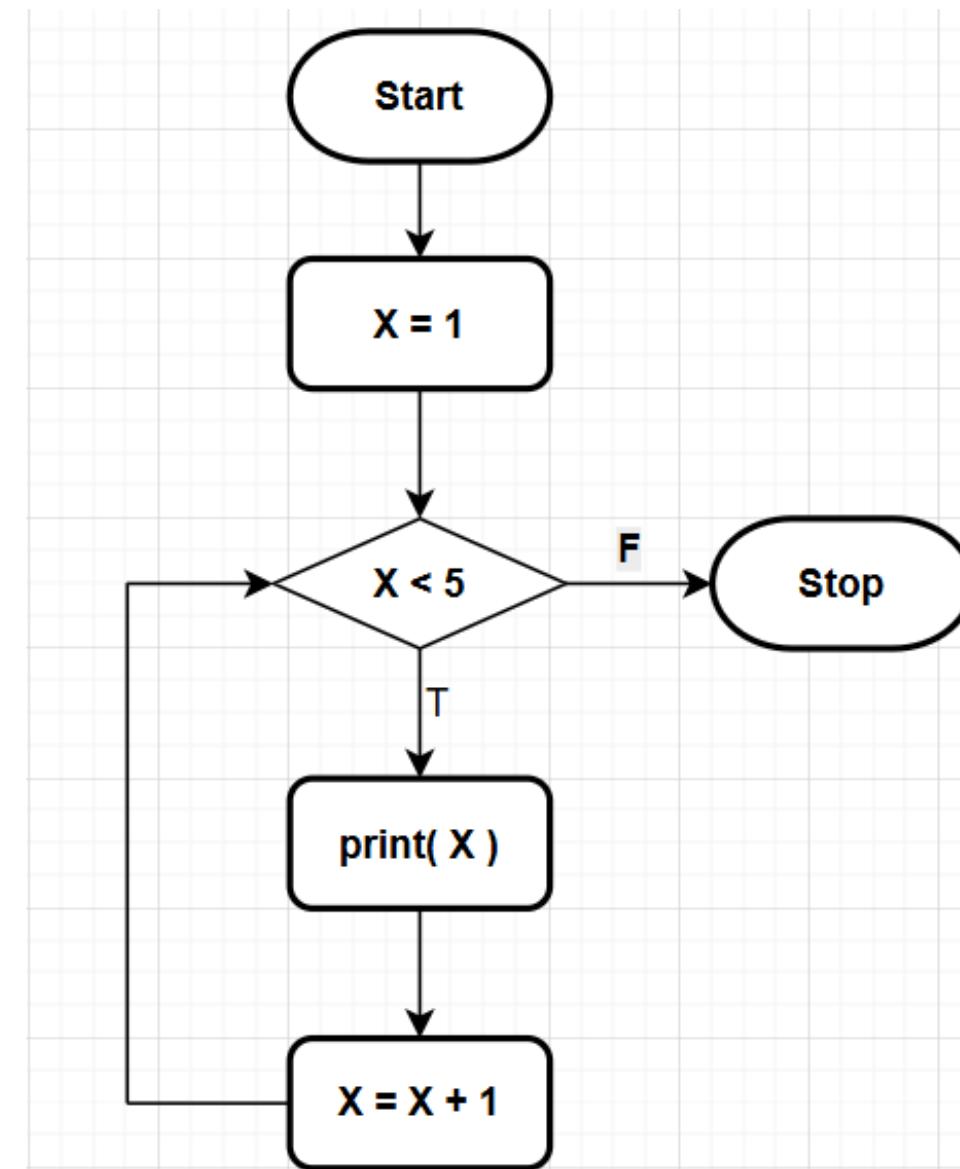
# Selection / Decision

- ทำงานเงื่อนไข

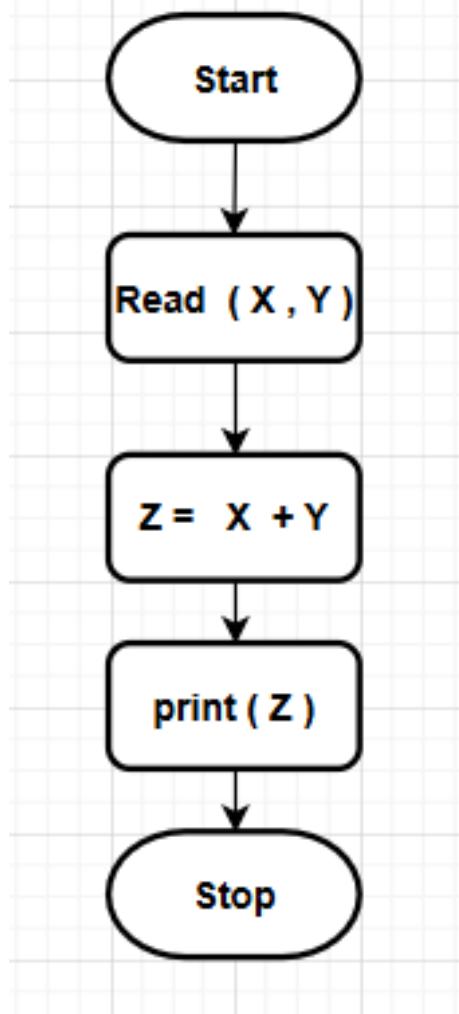


# Loop / Repetition

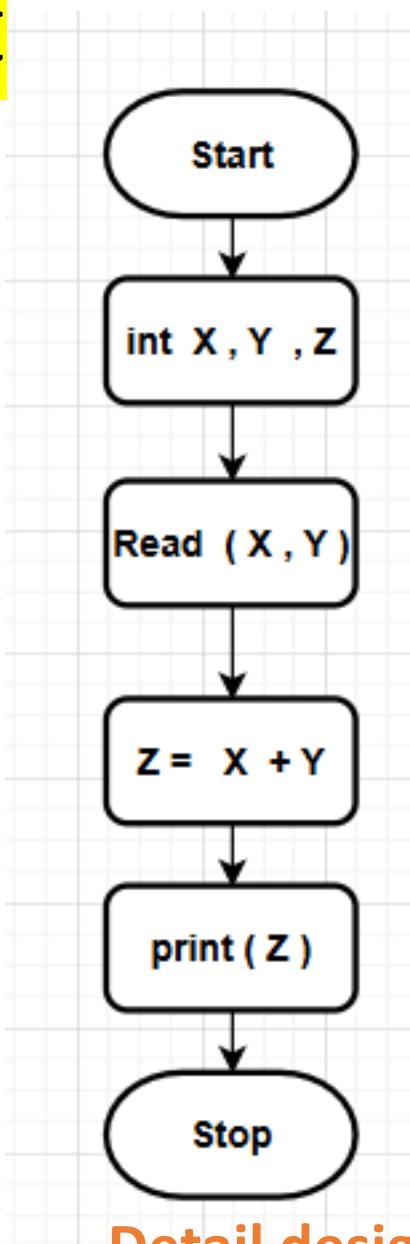
ทำซ้ำหลายครั้ง



# หลักการเขียน flowchart



concept

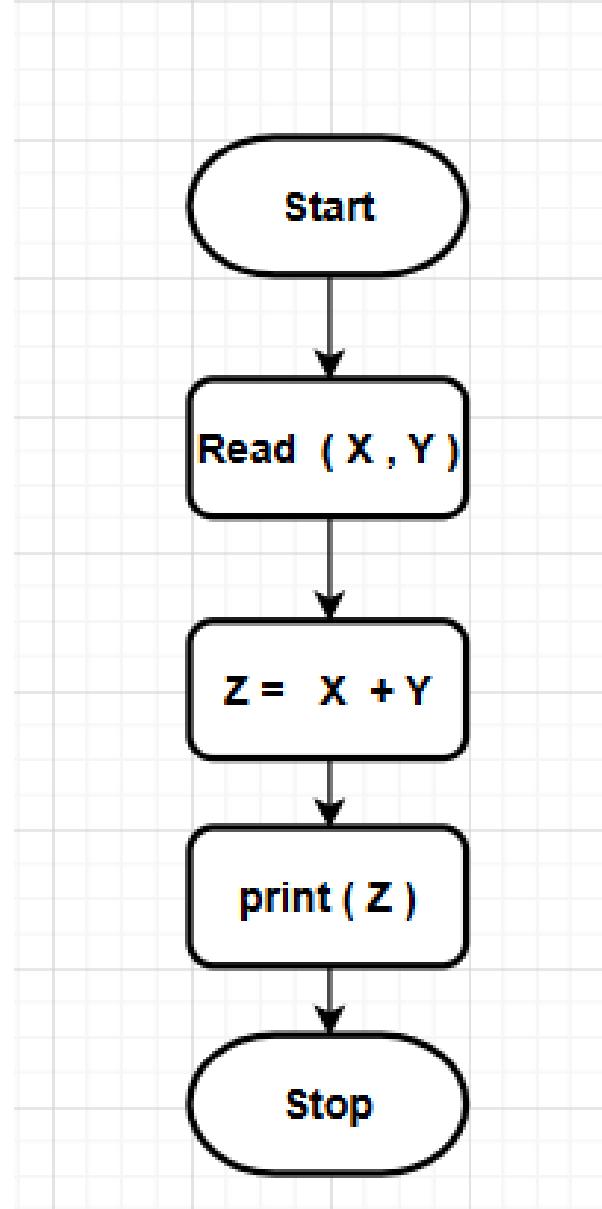
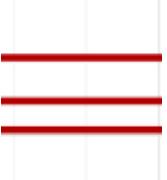
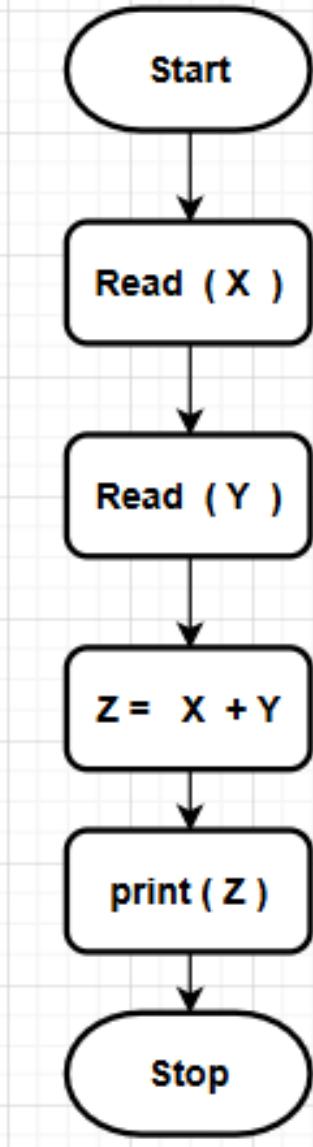


Detail design

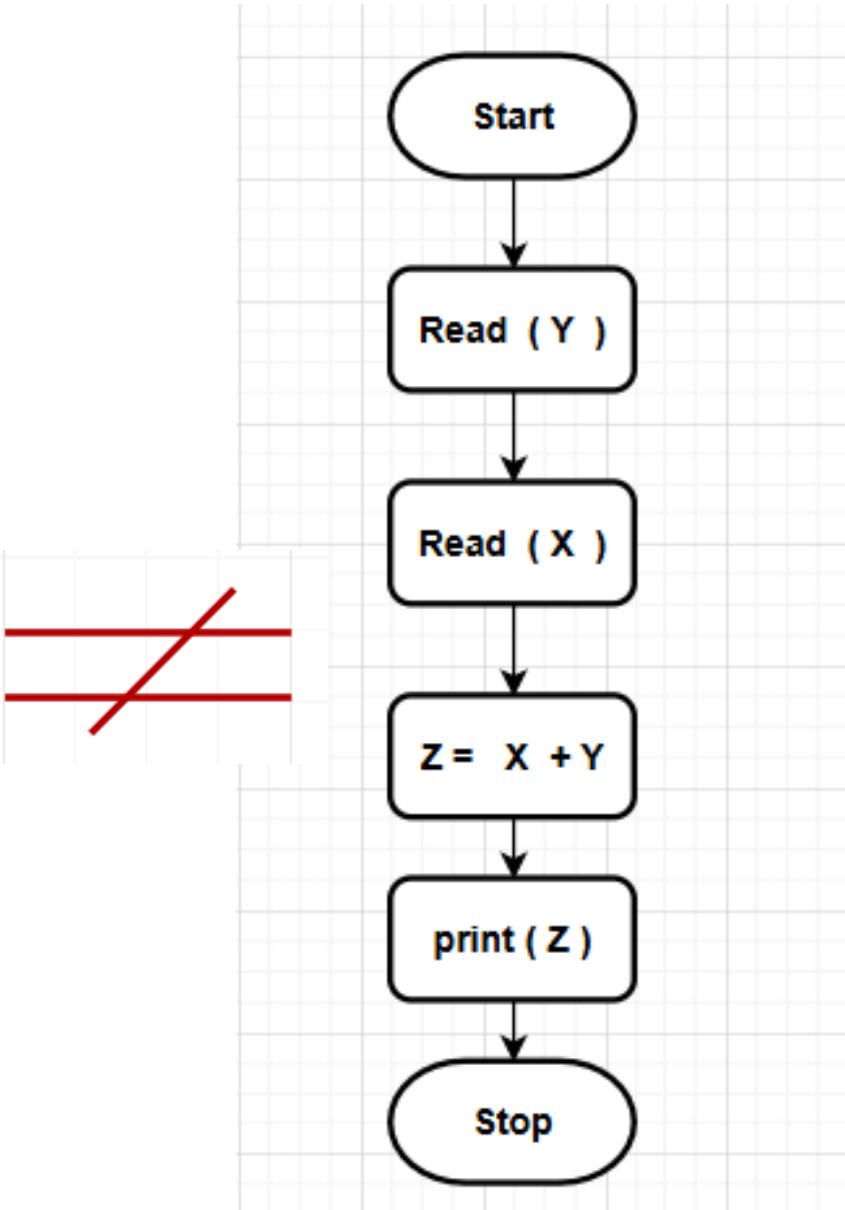
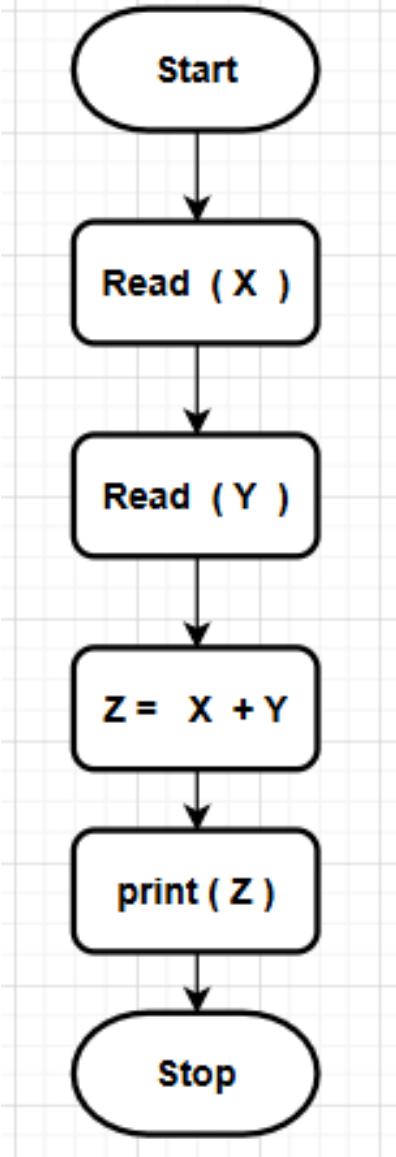
```
9 #include <iost
10 //addXY.CPP
11 int main()
12 {
13     int X,Y,Z;
14     printf( "READ X = ");
15     scanf("%d",&X);
16     printf( "READ Y = ");
17     scanf("%d",&Y);
18     Z = X + Y;
19     printf( "Z = %d",Z);
20
21     return 0;
22 }
```

The screenshot shows a code editor displaying a C++ program named "addXY.CPP". The code reads two integers (X and Y) from the user, calculates their sum (Z), and prints the result. The code uses standard input-output functions like printf and scanf.

# หลักการเขียน flowchart



# หลักการเขียน flowchart



END