|  |  |  |  |
| --- | --- | --- | --- |
| ลำดับ |  | สัญลักษณ์ | ความหมาย |
| 1 |  | Terminal | การเริ่มต้น หรือ การสิ้นสุด |
| 2 |  | Preparation | การกำหนดค่าต่างๆ ล่วงหน้า ซึ่งเป็นการทำงานภายในช่วงหนึ่งที่ซ้ำๆ กัน |
| 3 |  | Manual Input | การรับเข้าข้อมูลทางแป้นพิมพ์ |
| 4 |  | Show or Display | จอแสดงผล |
| 5 |  | Judgment or Decision | การตัดสินใจการเปรียบเทียบ จะมีผลใน 2 ทิศทาง คือ กรณีตรวจสอบเงื่อนไขเป็นเท็จและเป็นจริง |
| 6 |  | Data | รับ หรือ แสดงข้อมูลโดยไม่ระบุชนิดอุปกรณ์ เช่น อีเมล คำสั่งซื้อ รายงาน |
| 7 |  | Processing | การกำหนดค่า หรือการประมวลผลทั่วไป |
| 8 |  | Connector | จุดเชื่อมต่อจากจุดหนึ่งไปอีกจุดหนึ่งในหน้าเดียวกัน โดยไม่ต้องลากเส้นแต่อ้างอิงแทน |

## Flowchart

Flow chart (โฟลวชาร์ต) หรือผังงาน คือ แผนภาพที่มีการใช้สัญลักษณ์รูปภาพและลูกศรที่แสดงให้เห็นถึงขั้นตอนการทำงานของระบบทีละขั้นตอนอย่างชัดเจน โดยแต่ละสัญลักษณ์ในแผนภาพ Flowchart นั้นจะหมายถึงการทำงานในหนึ่งขั้นตอน สัญลักษณ์ลูกศรจะแทนลำดับการทำงานในขั้นตอนต่าง ๆ และยังแสดงให้เห็นถึงทิศทางการไหลของข้อมูล ตั้งแต่เริ่มต้นการทำงานจนกระทั่งได้ผลลัพธ์ตามวัตถุประสงค์ที่ได้ตั้งไว้ โดยการที่จะเขียนผังงานได้นั้น จะต้องสามารถวิเคราะห์ระบบการทำงานอย่างเป็นลำดับขั้นตอนก่อน จากนั้นจึงนำขั้นตอนเหล่านั้นมาเขียนในรูปแบบของผังงานหรือ Flowchart

## รูปแบบของผังงาน Flowchart

โครงสร้างของผังงาน หรือ Flowchart สามารถแบ่งได้เป็น  3  รูปแบบคือ การทำงานแบบตามลำดับ (Sequence Flowchart), การเลือกกระทำตามเงื่อนไข (Selection Flowchart), การทำซ้ำ (Iteration Flowchart) สามารถอธิบายโดยสังเขปได้ดังนี้

### **การทำงานแบบตามลำดับ (Sequence Flowchart)**

การทำงานแบบตามลำดับ (Sequence Flowchart) เป็นการเขียน Flowchart โดยจัดลำดับขั้นตอนการทำงานจากบนลงล่าง มีโครงสร้างในการเขียนคำสั่งเป็นบรรทัด และโดยทำงานตามคำสั่งทีละบรรทัดจากบรรทัดแรกที่เป็นการเริ่มต้นคำสั่งลงไปจนถึงบรรทัดล่างสุดเป็นการสิ้นสุดคำสั่ง

### **การทำงานแบบเลือกกระทำตามเงื่อนไข (Selection Flowchart)**

การทำงานแบบเลือกกระทำตามเงื่อนไข (Selection Flowchart) คือ การตัดสินใจหรือการเลือกทำตามเงื่อนไข โดยการวิเคราะห์จากข้อมูลจะมีเหตุการณ์ให้ดำเนินการต่อไป 2 กระบวนการ คือ ถ้าหากเงื่อนไขเป็นจริงจะดำเนินการตามกระบวนการหนึ่ง และหากเงื่อนไขเป็นเท็จจะดำเนินการอีกกระบวนการหนึ่ง

### **การทำซ้ำ (Iteration Flowchart)**

การทำซ้ำ (Iteration Flowchart) เป็นการเขียน flowchart ให้กลับมาทำงานในขั้นตอนแบบเดิมซ้ำ ๆ ซึ่งจะเห็นว่า flowchart มีขั้นตอนการทำงานบางขั้นตอนได้รับการประมวลผลมากกว่า 1 ครั้ง ซึ่งเรียกว่า loop โครงสร้างผังงานแบบทำซ้ำนี้จะมีการใช้ “สัญลักษณ์การตัดสินใจ”เข้ามาเปรียบเทียบเงื่อนไขเพื่อให้มีการทำงานซ้ำ โดยแบ่งได้ 3 ลักษณะ ดังนี้

1. **การทำซ้ำแบบลูป for** (for Loop) เป็นการทำซ้ำที่มีการกำหนดจำนวนรอบที่แน่นนอน โดยเริ่มต้นการเปรียบเทียบกับเงื่อนไข
2. **การทำซ้ำแบบลูป while**(while Loop) เป็นคำสั่งการทำซ้ำที่จะเริ่มด้วยการตรวจสอบเงื่อนไข หากเงื่อนไขเป็น “จริง” จะมีการทำซ้ำต่อไป แต่ถ้าหากเงื่อนไขเป็น “เท็จ” ก็จะเลิกดำเนินการทำตามคำสั่ง
3. **การทำซ้ำแบบลูป do…while** (do…while Loop) เป็นคำสั่งการทำซ้ำที่จะเริ่มด้วยการทำงาน 1 รอบ แล้วจึงค่อยตรวจสอบเงื่อนไข ถ้าหากเงื่อนไขเป็น “จริง” ก็จะมีการทำซ้ำต่อ ถ้าเงื่อนไขเป็น “เท็จ” ก็จะเลิกดำเนินการทำคำสั่ง

เรื่องตัวแปรในภาษา C เป็นพื้นฐานที่สำคัญมาก ผมจะอธิบายให้เข้าใจง่ายที่สุดดังนี้ครับ

**ตัวแปรคืออะไร?**

ในภาษา C (และภาษาโปรแกรมอื่นๆ) ตัวแปรเปรียบเสมือน **"กล่อง"** ที่ใช้สำหรับ **เก็บข้อมูล** ชั่วคราวในหน่วยความจำของคอมพิวเตอร์ ข้อมูลที่เก็บไว้ในตัวแปรสามารถเปลี่ยนแปลงได้ระหว่างการทำงานของโปรแกรม

**ทำไมต้องมีตัวแปร?**

เราใช้ตัวแปรเพื่อ:

* **เก็บข้อมูลที่ผู้ใช้ป้อนเข้ามา:** เช่น ชื่อ, อายุ, หมายเลขโทรศัพท์
* **เก็บผลลัพธ์จากการคำนวณ:** เช่น ผลรวมของตัวเลข, ค่าเฉลี่ย
* **เก็บสถานะต่างๆ ของโปรแกรม:** เช่น ตัวนับรอบ, แฟล็ก (flag) ที่บ่งบอกเงื่อนไข
* **อ้างอิงถึงข้อมูลได้ง่าย:** แทนที่จะต้องจำตำแหน่งของข้อมูลในหน่วยความจำโดยตรง เราสามารถใช้ชื่อตัวแปรที่สื่อความหมายแทนได้

**การประกาศตัวแปร (Declaration)**

ก่อนที่จะใช้งานตัวแปรได้ เราต้องทำการ **ประกาศ (declare)** ตัวแปรเสียก่อน การประกาศตัวแปรคือการบอกให้คอมพิวเตอร์ทราบว่าเราต้องการสร้าง "กล่อง" ชื่ออะไร และจะให้ "กล่อง" นี้เก็บข้อมูลประเภทใด

**รูปแบบการประกาศตัวแปร:**

data\_type variable\_name;

* **data\_type**: คือ **ชนิดของข้อมูล** ที่ตัวแปรนี้จะสามารถเก็บได้ เช่น จำนวนเต็ม, เลขทศนิยม, ตัวอักษร
* **variable\_name**: คือ **ชื่อของตัวแปร** ที่เราตั้งขึ้น เพื่อใช้อ้างอิงถึงข้อมูลที่เก็บอยู่ในนั้น

**ตัวอย่างการประกาศตัวแปร:**

int age; // ประกาศตัวแปรชื่อ age สำหรับเก็บจำนวนเต็ม

float salary; // ประกาศตัวแปรชื่อ salary สำหรับเก็บเลขทศนิยม

char initial; // ประกาศตัวแปรชื่อ initial สำหรับเก็บตัวอักษร 1 ตัว

int count, total; // ประกาศตัวแปรหลายตัวพร้อมกัน (count และ total) สำหรับเก็บจำนวนเต็ม

**การกำหนดค่าให้กับตัวแปร (Initialization)**

หลังจากประกาศตัวแปรแล้ว เราสามารถ **กำหนดค่า (initialize)** หรือใส่ข้อมูลเริ่มต้นให้กับตัวแปรได้

**รูปแบบการกำหนดค่าให้กับตัวแปร:**

variable\_name = value;

* **variable\_name**: คือชื่อของตัวแปรที่เราได้ประกาศไว้
* **value**: คือค่าที่เราต้องการเก็บไว้ในตัวแปรนั้น

**เราสามารถประกาศและกำหนดค่าให้กับตัวแปรพร้อมกันได้:**

int age = 30; // ประกาศตัวแปร age เป็นชนิด int และกำหนดค่าเริ่มต้นเป็น 30

float salary = 25000.50; // ประกาศตัวแปร salary เป็นชนิด float และกำหนดค่าเริ่มต้นเป็น 25000.50

char initial = 'J'; // ประกาศตัวแปร initial เป็นชนิด char และกำหนดค่าเริ่มต้นเป็น 'J'

**ชนิดข้อมูลพื้นฐานในภาษา C (Data Types)**

ภาษา C มีชนิดข้อมูลพื้นฐานที่สำคัญดังนี้:

* **int (integer):** สำหรับเก็บจำนวนเต็ม (ไม่มีทศนิยม) เช่น -5, 0, 100
* **float (floating-point):** สำหรับเก็บเลขทศนิยมที่มีความแม่นยำระดับหนึ่ง เช่น 3.14, -2.5
* **double (double-precision floating-point):** สำหรับเก็บเลขทศนิยมที่มีความแม่นยำสูงกว่า float
* **char (character):** สำหรับเก็บตัวอักษร 1 ตัว เช่น 'a', 'Z', '9'
* **void:** ใช้ในกรณีที่ฟังก์ชันไม่มีการส่งค่ากลับ หรือเป็นตัวชี้ (pointer) ที่สามารถชี้ไปยังข้อมูลชนิดใดก็ได้
* **short int, long int, long long int, unsigned int, signed int, unsigned char, signed char:** เป็นชนิดข้อมูลที่ปรับปรุงมาจากชนิดพื้นฐาน เพื่อกำหนดขนาดของข้อมูลและช่วงของค่าที่สามารถเก็บได้

**กฎการตั้งชื่อตัวแปร (Naming Rules)**

ในการตั้งชื่อตัวแปรในภาษา C มีกฎเกณฑ์ดังนี้:

* ชื่อตัวแปรต้องประกอบด้วย ตัวอักษร (a-z, A-Z), ตัวเลข (0-9), และเครื่องหมายขีดล่าง (\_) เท่านั้น
* ตัวอักษรตัวแรกของชื่อตัวแปรต้องเป็นตัวอักษรหรือเครื่องหมายขีดล่าง (\_) เท่านั้น (ห้ามเป็นตัวเลข)
* ชื่อตัวแปรมีความยาวได้ แต่คอมไพเลอร์บางตัวอาจมีข้อจำกัดเรื่องความยาว
* ภาษา C เป็น **case-sensitive** หมายความว่า age, Age, และ AGE ถือเป็นตัวแปรที่แตกต่างกัน
* ห้ามใช้คำสงวน (reserved words) ของภาษา C เป็นชื่อตัวแปร (เช่น int, float, if, else, while, for, ฯลฯ)
* ควรตั้งชื่อตัวแปรให้สื่อความหมาย เพื่อให้โค้ดอ่านง่ายและเข้าใจได้ง่าย

**ขอบเขตของตัวแปร (Scope)**

ขอบเขตของตัวแปรหมายถึง บริเวณของโค้ดที่ตัวแปรนั้นสามารถเข้าถึงและใช้งานได้ หลักๆ มี 2 ประเภท:

* **ตัวแปรเฉพาะที่ (Local Variable):** ถูกประกาศภายในบล็อกของโค้ด (เช่น ภายในฟังก์ชัน หรือภายในลูป) ตัวแปรเหล่านี้จะสามารถใช้งานได้เฉพาะภายในบล็อกนั้นเท่านั้น เมื่อออกจากบล็อก ตัวแปรเหล่านี้จะถูกทำลาย
* **ตัวแปรส่วนกลาง (Global Variable):** ถูกประกาศภายนอกฟังก์ชันใดๆ ตัวแปรเหล่านี้สามารถใช้งานได้จากทุกส่วนของโปรแกรม

**อายุของตัวแปร (Lifetime)**

อายุของตัวแปรหมายถึง ระยะเวลาที่ตัวแปรนั้นยังคงมีอยู่ในหน่วยความจำ

* **ตัวแปรเฉพาะที่ (Local Variable):** มีอายุการใช้งานเฉพาะในช่วงเวลาที่บล็อกของโค้ดที่ตัวแปรนั้นถูกประกาศกำลังทำงาน เมื่อบล็อกนั้นสิ้นสุดลง ตัวแปรก็จะถูกทำลาย
* **ตัวแปรส่วนกลาง (Global Variable):** มีอายุการใช้งานตลอดระยะเวลาที่โปรแกรมทำงาน

**ตัวอย่างโค้ดที่แสดงการใช้งานตัวแปร:**

#include <stdio.h>

int main() {

int num1 = 10;

int num2 = 5;

int sum;

sum = num1 + num2;

printf("ผลรวมของ %d และ %d คือ %d\n", num1, num2, sum);

float average = (float)sum / 2; // ต้อง cast sum เป็น float เพื่อให้ผลลัพธ์เป็นทศนิยม

printf("ค่าเฉลี่ยคือ %.2f\n", average); // แสดงผลลัพธ์ทศนิยม 2 ตำแหน่ง

char grade = 'A';

printf("เกรดที่ได้คือ %c\n", grade);

return 0;

}