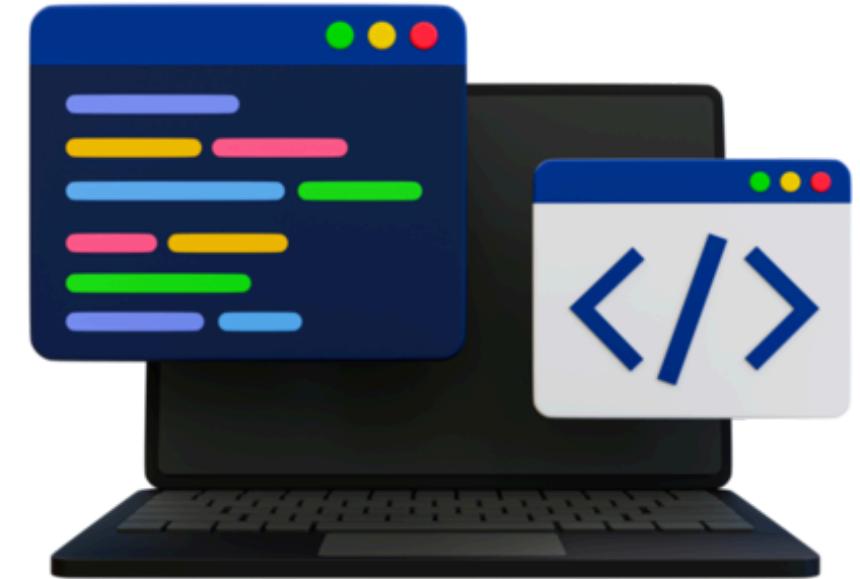


WELCOM TO C CODING

BY THANAPHOL RENGRAD



CONTENTS

- HISTORY OF C LANGUAGE**
- FUNDAMENTAL OF C LANGUAGE**
- INSTALL DEV C++**
- VARIABLES**
- OPERATORS**
- CONDITIONS**
- LOOPING**
- WORK SHOPS**



HISTORY OF C LANGUAGE

ประวัติของภาษา C

ภาษาซี (C) เป็นภาษาโปรแกรมระดับกลางที่ถูกพัฒนาขึ้นในช่วงต้นทศวรรษ 1970 โดย Dennis Ritchie ที่ Bell Labs ภาษาซีเป็นที่รู้จักกันดีในเรื่องของความยืดหยุ่นและประสิทธิภาพ ซึ่งเหมาะสมสำหรับการพัฒนาโปรแกรมในระบบที่หลากหลาย เช่น ระบบปฏิบัติการ ตัวแปลภาษา และโปรแกรมที่ต้องการประสิทธิภาพสูง



FUNDAMENTAL OF C LANGUAGE

พื้นฐานของภาษา C

โครงสร้างของภาษา C หลักๆ เลยจะมีในส่วนของ

- Header File หรือ ไฟล์เก็บชุดคำสั่งต่างๆ ที่ใช้ในโปรแกรม
- Main Function หรือ พังก์ชันที่ใช้ในการประมวลผลชุดคำสั่งที่เราเขียนโปรแกรม
- ทุกครั้งที่เขียนจบบรรทัดต้องมีเครื่องหมาย semi colon (;)



HEADER FILE

ไฟล์ที่เก็บชุดคำสั่ง

จุดสังเกต

- อญูข้างบนสุดของบรรทัด
- มีนามสกุล file คือ (.h) หรือ จุด h เช่น stdio.h

```
1 #include<stdio.h>
2 int a = 1;
3 int b = 3;
```



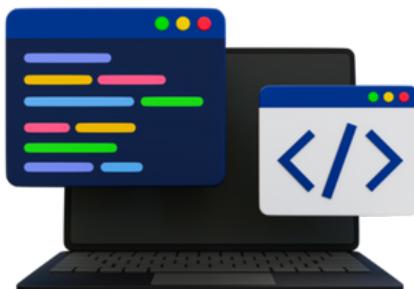
MAIN FUNCTION

ฟังก์ชันหลัก

จุดสังเกต

- จะมีคำว่า main ตามด้วยเครื่องหมายวงเล็บ เช่น main()

```
5 main()
6 {
7     c=a+b;
8     printf("Sum =%d",c);
9 }
```



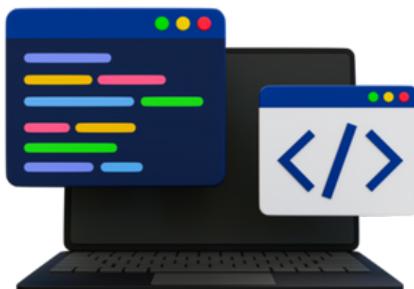
MAIN FUNCTION

ฟังก์ชันหลัก

จุดสังเกต

- จะมีคำว่า main ตามด้วยเครื่องหมายวงเล็บ เช่น main()

```
5 main()
6 {
7     c=a+b;
8     printf("Sum =%d",c);
9 }
```



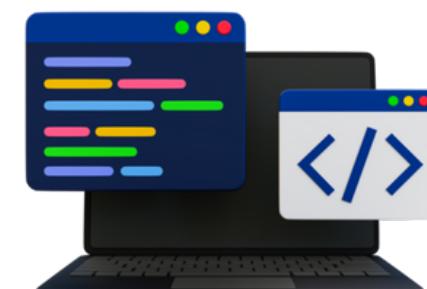
EXAMPLE OF C STRUCTURE

โครงสร้างของภาษา C

```
1 #include<stdio.h>
2 int a = 1;
3 int b = 3;
4 int c;
5 main()
6 {
7     c=a+b;
8     printf("Sum =%d
9 }
```

คำถ้า

1. บรรทัดที่ 1 เรียกว่าอะไร ?
2. บรรทัดที่ 5 เรียกว่าอะไร ?



DATA TYPE

ชนิดของข้อมูล

ชนิดของข้อมูลหลักๆ ในการเขียนโปรแกรม สามารถแบ่งออกได้ดังนี้

- ชนิดข้อมูลที่เก็บตัวเลข ในทางการเขียนโปรแกรมแบ่งออกเป็น 2 กรณี ดังนี้
 - ตัวเลขที่มี ทศนิยม ---> float , double เป็นต้น
 - ตัวเลขที่ไม่มี ทศนิยม ---> int เป็นต้น
- ชนิดข้อมูลที่เก็บตัวอักษร ในทางการเขียนโปรแกรมแบ่งออกเป็น 2 กรณี ดังนี้
 - ตัวอักษรเพียงตัวเดียว --> char เป็นต้น
 - ตัวอักษรหลายตัว --> char [จำนวนตัวอักษร] , string เป็นต้น

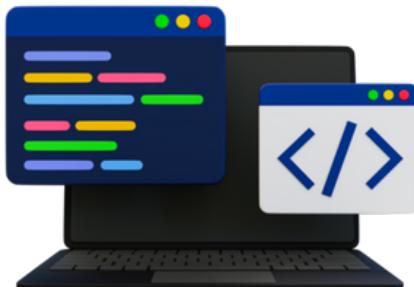


DATA TYPE

ชนิดของข้อมูล

ชนิดของข้อมูลหลักๆ ในการเขียนโปรแกรม สามารถแบ่งออกได้ดังนี้

- ชนิดข้อมูลที่เก็บค่าความเป็น จริง เท็จ(ไม่จริง)
 - ข้อมูลชนิดนี้เรียกว่า ข้อมูลแบบ boolean มีแค่ค่า --> true และ false



VARIABLES

ตัวแปร

วิธีการประกาศตัวแปร สามารถประกาศได้ดังนี้

TYPE VARIABLE_NAME = VALUE

ชนิดตัวแปร ชื่อตัวแปร = ค่าของตัวแปร

ตัวอย่าง

```
INT X = 20;  
CHAR NAME = "THANAPHOL";
```

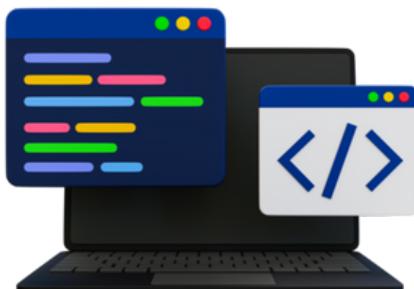


OPERATORS

ตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการคณิตศาสตร์ในการเขียนโปรแกรม มี 5 ตัวดังนี้

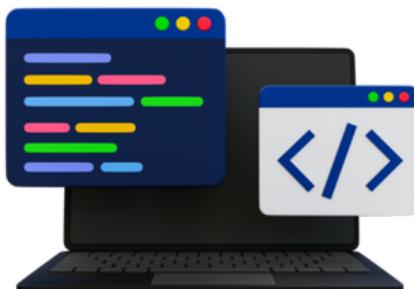
- บวก (+)
- ลบ (-)
- คูณ (*)
- หารเอาแค่ผลลัพธ์ (/)
- หารเอาแค่ตัวเศษ (%)



EXAMPLE

ตัวอย่าง

- $5 + 2 = 7$
- $5 - 2 = 3$
- $5 * 2 = 10$
- $5 / 2 = 2$
- $5 \% 2 = 1$

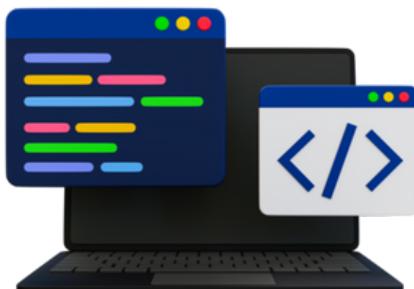


CONDITIONS

เงื่อนไข

ในการเขียนโปรแกรม เราจะใช้การเช็คเงื่อนเพื่อควบคุมการทำงานของโปรแกรม โดยจะใช้คำสั่งดังนี้

- if
- else



CONDITIONS

เงื่อนไข

- ถ้าเงื่อนไขที่เช็คเป็นจริง code ก็จะทำงานในส่วนของ if
- ถ้าเงื่อนไขที่เช็คเป็นเท็จ code ก็จะทำงานในส่วนของ else

```
#include <stdio.h>

int main()
{
    int a = 5;
    int b = 10;

    if (a == b)
    {
        printf ("a and b are equal");
    }
    else
    {
        printf ("a and b are not equal");
    }

    return 0;
}
```



EXAMPLE

ตัวอย่าง

```
#include <stdio.h>

int main()
{
    int a = 5;
    int b = 10;

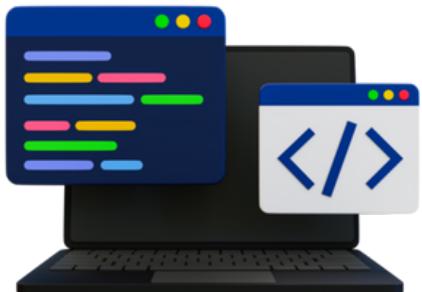
    if (a == b)
    {
        printf ("a and b are equal");
    }
    else
    {
        printf ("a and b are not equal");
    }

    return 0;
}
```



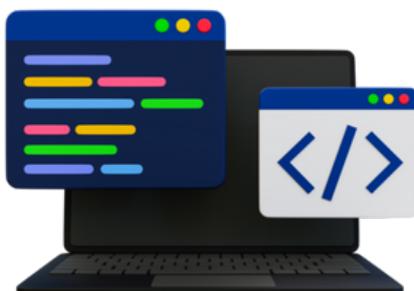
LOOPING

แน่นอนว่าการเขียนโปรแกรมด้วยชุดคำสั่งเดิมๆ ซ้ำ มักจะทำให้เกิดความยุ่งยากในการเขียนโปรแกรม การที่เลือกใช้วิธีการวนซ้ำหรือ Looping ก็จะเข้ามาช่วยในการทำงานให้มีประสิทธิภาพดีขึ้น



EXAMPLE

แน่นอนว่าการเขียนโปรแกรมด้วยชุดคำสั่งเดิมๆ ช้า มักจะทำให้เกิดความยุ่งยากในการเขียนโปรแกรม การที่เลือกใช้วิธีการวนซ้ำหรือ Looping ก็จะเข้ามาช่วยในการทำงานให้มีประสิทธิภาพดีขึ้น



EXAMPLE

Loop แบ่งออกเป็น 3 แบบ

- While Loop
- Do While Loop
- For Loop



WHILE LOOP

ใช้คำสั่ง While Loop ใช้ในการนี่ที่เราไม่รู้การวนซ้ำหรือจำนวนรอบที่ชัดเจน

```
while(เงื่อนไข) {  
    คำสั่งด้านในจะถูกวนซ้ำ  
}
```

ถ้าเงื่อไขเป็นจริงจะมีการวนซ้ำ
ตลอด ไม่มีที่สิ้นสุด หรือที่เรียกว่า Infinite Loop



EXAMPLE WHILE LOOP

CODE

```
int i = 1 ;  
  
while(i < 5) {  
    printf("Hello World");  
}  
}
```

RESULT

```
Hello World ;  
Hello World ;  
Hello World ;  
Hello World ;  
Hello World ;
```

ไปเรื่อยๆไม่มีที่สิ้นสุด Infinite Loop



EXAMPLE WHILE LOOP

วิธีการแก้ Infinite Loop คือการเพิ่มค่าตัวแปรในเงื่อนไข เช่น

CODE

```
int i = 1 ;  
  
while(i < 5) {  
    printf("Hello World");  
    i + 1 ;  
}
```

RESULT

```
Hello World  
Hello World  
Hello World  
Hello World
```

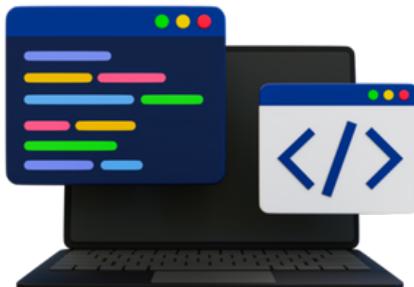


FOR LOOP

ใช้คำสั่ง For Loop ใช้ในกรณีที่เรารู้การวนซ้ำหรือจำนวนรอบที่ชัดเจน เช่น
อยากรูปสัก 5 รอบ 10 รอบ ก็สามารถกำหนดได้

```
int i;
```

```
For(i = จำนวนเลขที่ต้องการกำหนดให้ i ; i < จำนวนที่อยากรูป ; การอัพเดตค่า i) {  
    คำสั่งที่ต้องการให้ Loop  
}
```



EXAMPLE FOR LOOP

CODE

```
int i ;  
  
For(i = 0 ; i < 5; i+ 1) {  
    printf("Hello World") ;  
}
```

RESULT

```
Hello World  
Hello World  
Hello World  
Hello World  
Hello World
```



THANK YOU

GOOD LUCK

