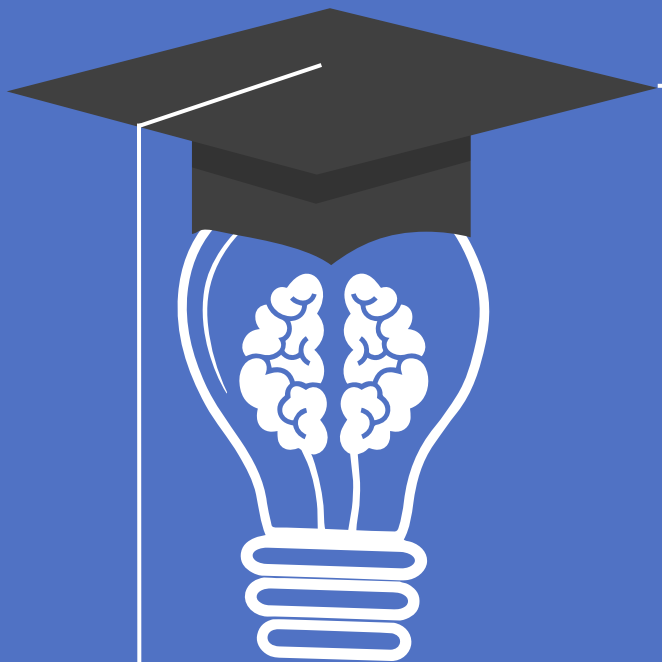


Language 
Step by step

ปุ่มต่าง ๆ ในโปรแกรม

```
1  #include <stdio.h>
2  int main()
3  {
4      printf("Answer is plus : 250 + 43 = %d\n", 250 + 43);
5      printf("Answer is minus: 250 - 50 = %d\n", 250 - 50);
6      printf("Answer is minus: 250.50- 50.50 = %.2f", 250.50 - 50.50);
7
8      return 0;
9  }
10
```



```
File Edit Selection View Go Run Terminal Help calculate.c - Solve  
C calculate.c X  
LearnC > C calculate.c > main()  
You, 9 hours ago | 1 author (You)  
1 #include <stdio.h>  
2 int main()  
3 {  
4     printf("Answer is plus : 250 + 43 = %d\n", 250 + 43);  
5     printf("Answer is minus: 250 - 50 = %d\n", 250 - 50);  
6     printf("Answer is minus: 250.50- 50.50 = %.2f", 250.50 - 50.50);  
7  
8     return 0;  
9 }  
10
```

เขียน Code ในส่วนนี้



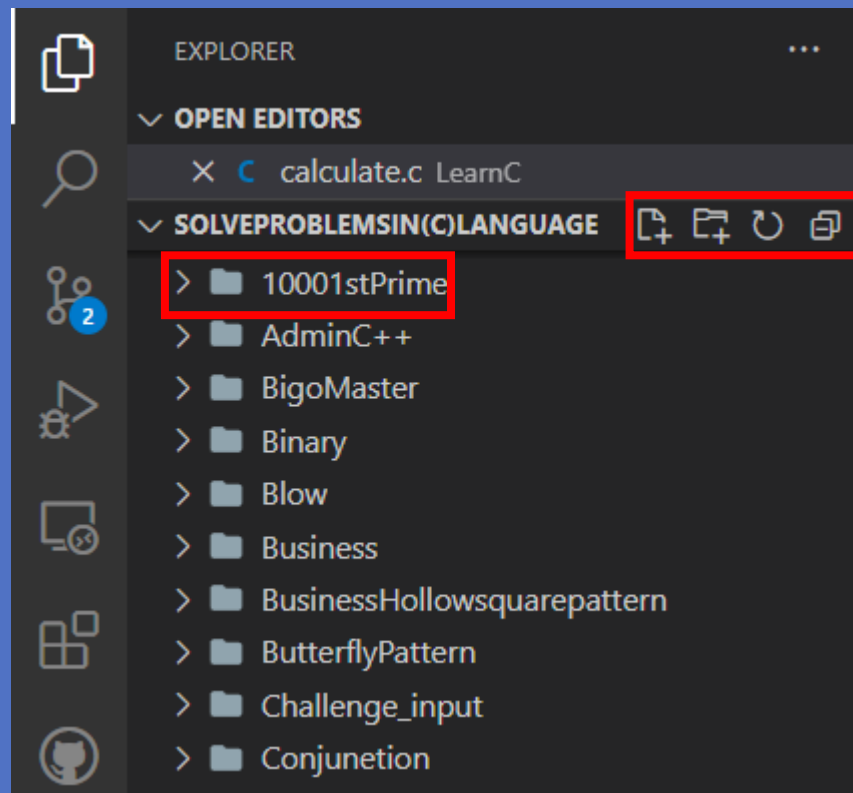
สร้างไฟล์
และ
เปิดไฟล์งาน

```
File Edit Selection View Go Run Terminal Help
calculate.c - Solve

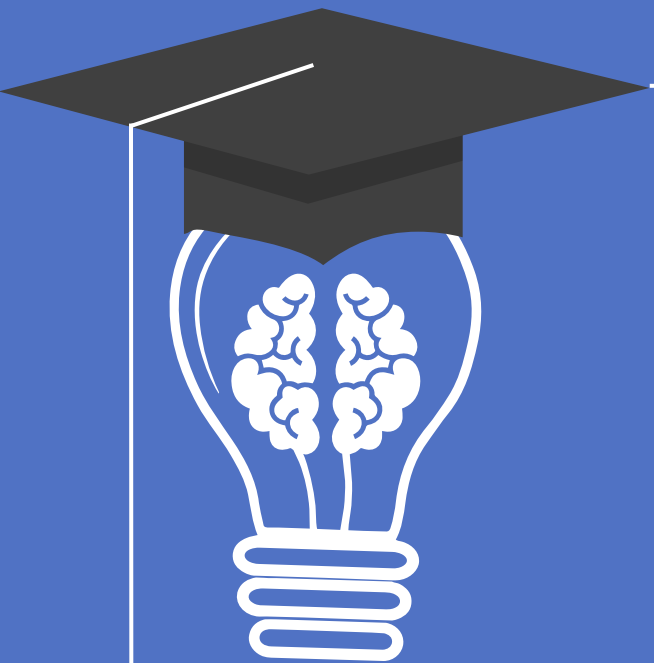
C calculate.c X
LearnC > C calculate.c > main()
You, 9 hours ago | 1 author (You)
1 #include <stdio.h>
2 int main()
3 {
4     printf("Answer is plus : 250 + 43 = %d\n", 250 + 43);
5     printf("Answer is minus: 250 - 50 = %d\n", 250 - 50);
6     printf("Answer is minus: 250.50- 50.50 = %.2f", 250.50 - 50.50);
7
8     return 0;
9 }
10
```



ชื่อโฟลเดอร์



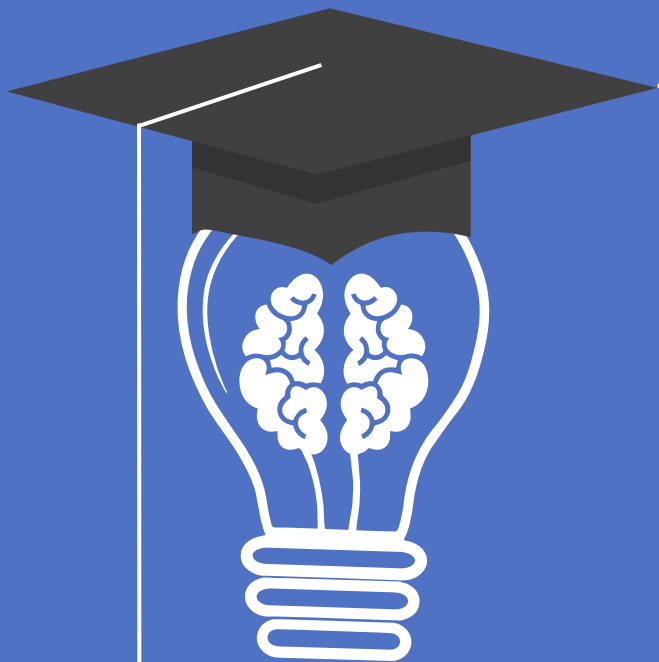
สร้างโฟลเดอร์
และสร้างไฟล์



File Edit Selection View Go **Run** **Terminal** Help

Run Code

```
calculate.c X [Run] [Settings] [Previous] [Next] [Search] [Run and Debug] [More]
LearnC > calculate.c > ...
You, 9 hours ago | 1 author (You)
1  #include <stdio.h>
2  int main()
3  {
4      printf("Answer is plus : 250 + 43 = %d\n", 250 + 43);
5      printf("Answer is minus: 250 - 50 = %d\n", 250 - 50);
6      printf("Answer is minus: 250.50- 50.50 = %.2f", 250.50 - 50.50);
7
8      return 0;
9  }
10
```

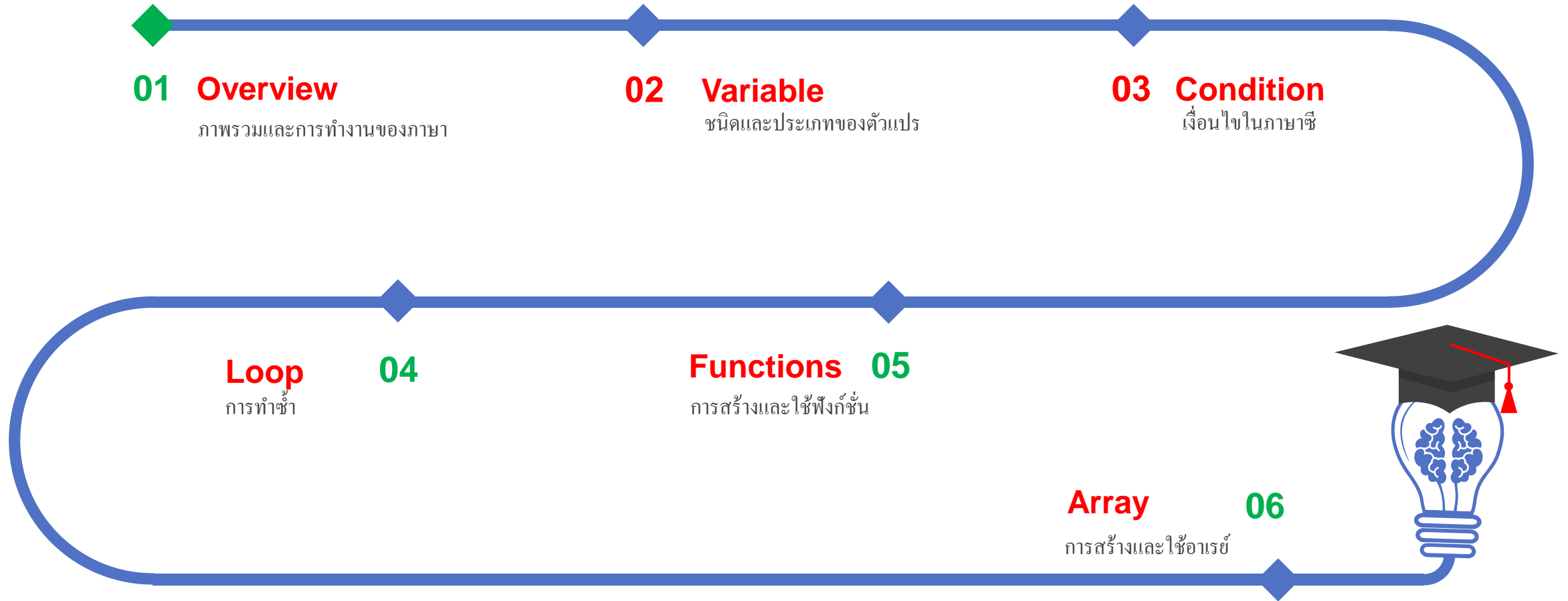


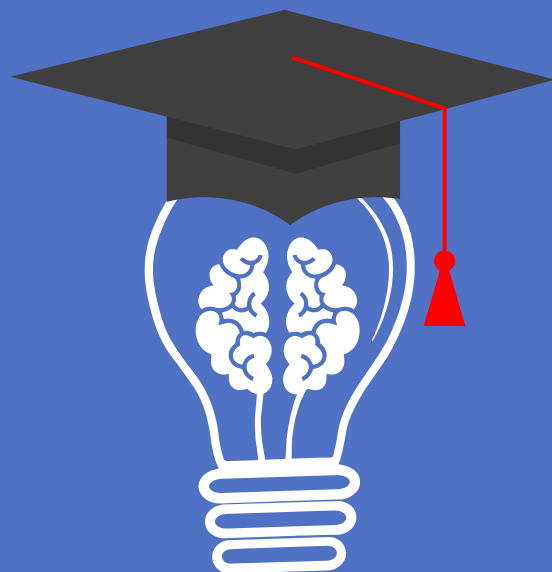
```
GITLENS  PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER  + v ^ x  
engine-In-21qcsb1b.s4s --stdout=Microsoft-MIEngine-Out-pqd  
klqew.frs --stderr=Microsoft-MIEngine-Error-gsuiqmb3.ial  
--pid=Microsoft-MIEngine-Pid-zjy120pk.pny --dbgExe=C:\Min  
GW\bin\gdb.exe --interpreter=mi "  
C:\Users\007xfa\Desktop\SolveProblemsIn(C)Language>
```

☒ cmd
☒ Code
☒ Code
☒ C/C++: ... ✓
☒ cppdbg: calc...

ผลการทำงานของโปรแกรม

Timeline Style





Overview

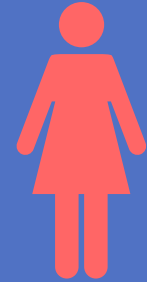
โครงสร้างของภาษา



Computer



Compiler



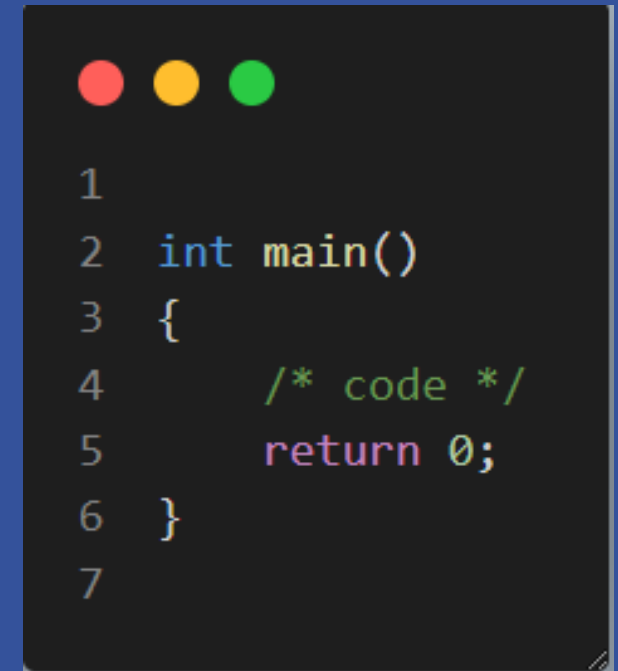
Person



Machine language

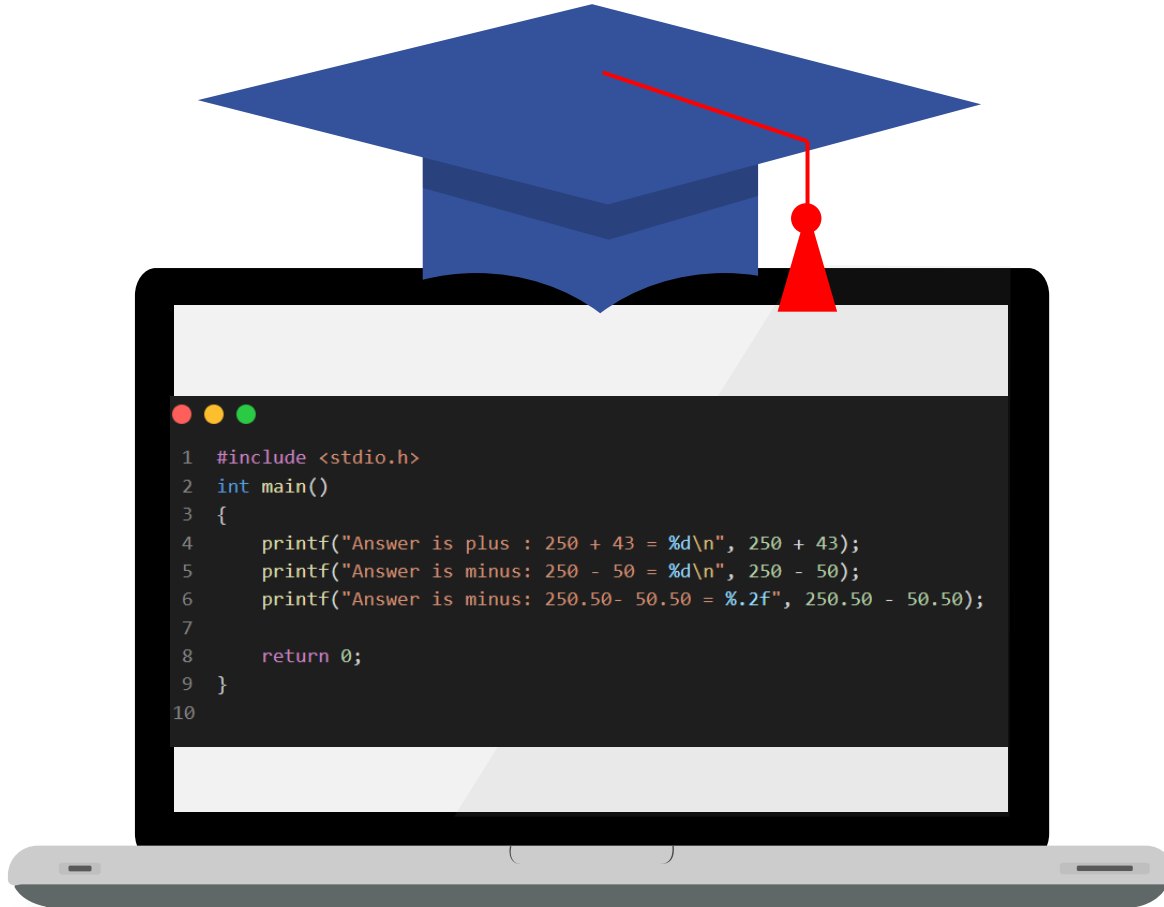


Text file



C language

Section



Comment



New line

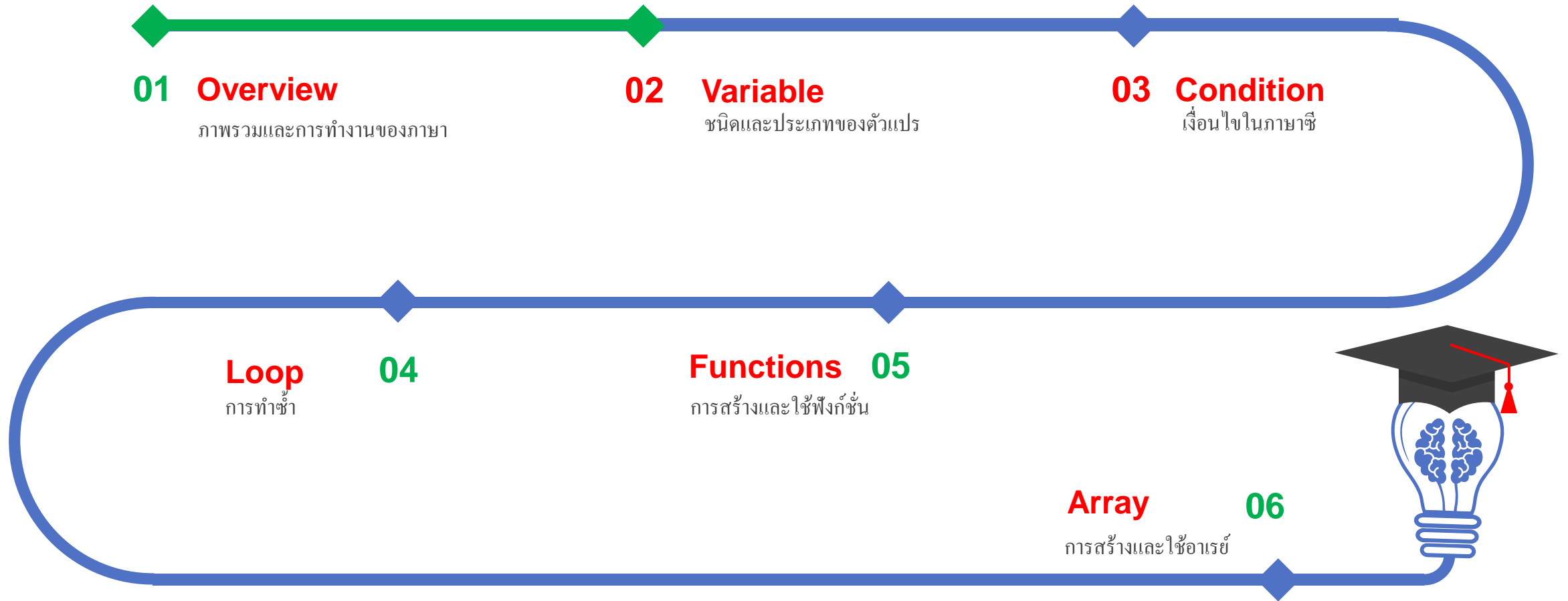


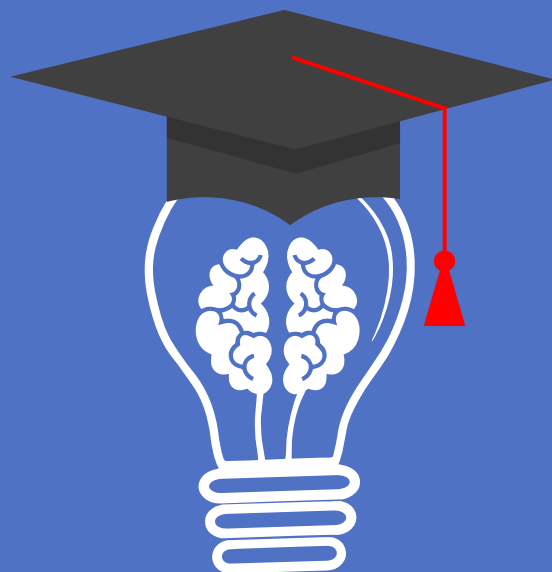
Horizontal tab



Main ()

Timeline Style





Variable

ชนิดและประเภทของตัวแปร

Variable



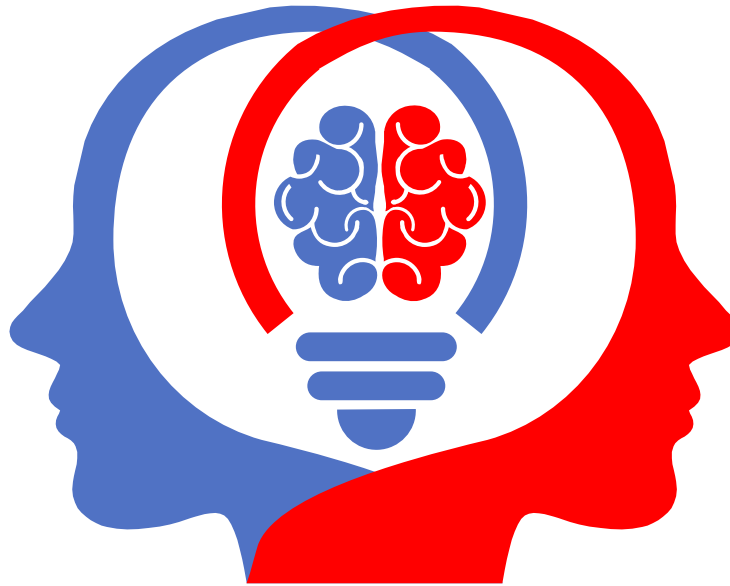
Integer Variable

ตัวแปรชนิดที่เก็บ จำนวนเต็ม



Float Variable

ตัวแปรชนิดที่เก็บ
จำนวนทศนิยม



Character Variable

ตัวแปรชนิดที่เก็บ ตัวอักษร



Variable Naming Rules

กฎการตั้งชื่อตัวแปร

Integer Variable

แบ่งออกเป็น 2 ประเภท

- int หรือ short
- long

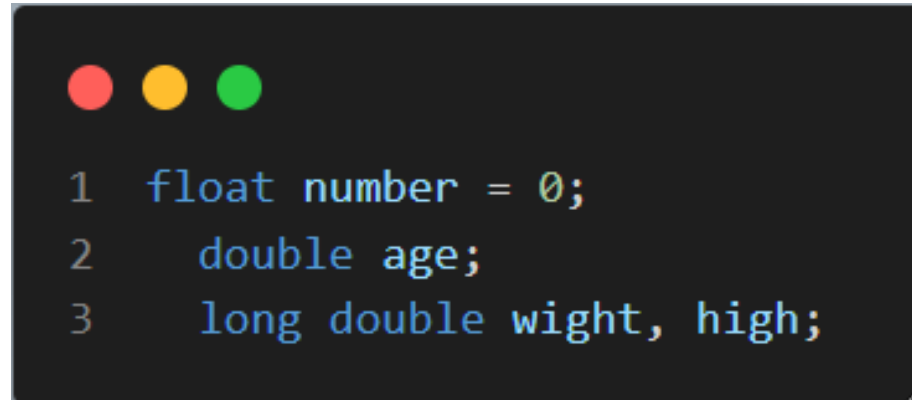
```
1 #include <stdio.h>
2 int main()
3 {
4     int number = 0;
5     int age;
6     int wight, high;
7     return 0;
8 }
```

ชนิด	ขนาดความกว้าง	ช่วงของค่า
int	2 byte	- 32,768 – 32,767
short	2 byte	- 32,768 – 32,767
long	4 byte	-2,147,483,648 - 2,147,483,647

Float Variable

แบ่งออกเป็น 3 ประเภท

- float
- double
- long double



```
1 float number = 0;
2 double age;
3 long double wight, high;
```

ชนิด	ขนาดความกว้าง	ช่วงของค่า
Float	4 byte	$3.4 \times 10^{-38} - 3.4 \times 10^{38}$
Double	8 byte	$1.7 \times 10^{-308} - 1.7 \times 10^{308}$
Long Double	10 byte	$3.4 \times 10^{-4932} - 3.4 \times 10^{4932}$

Character Variable

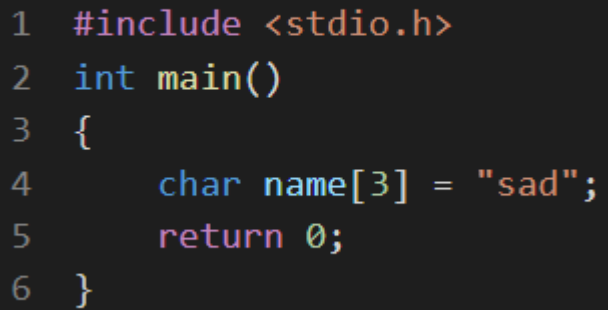
แบ่งออกเป็น 2 ประเภท

- char
- unsigned char

```
1 int main()
2 {
3     char nickName = 'f';
4     unsigned char name = 'f';
5
6     return 0;
7 }
```

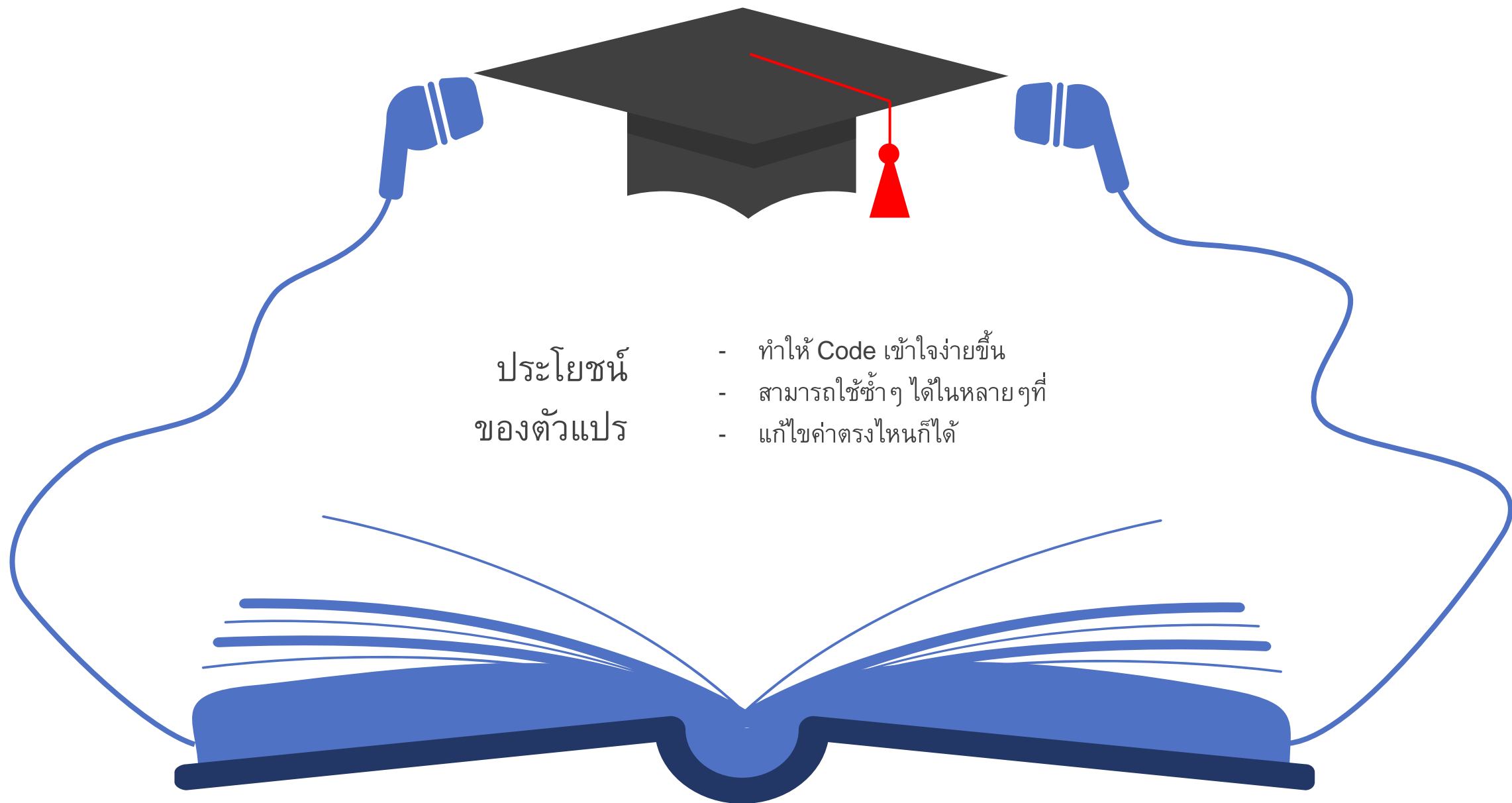
ชนิด	ขนาดความกว้าง	ช่วงของค่า
Char	1 byte	-128 - 127
Unsigned Char	1 byte	0 - 255

Character Variable



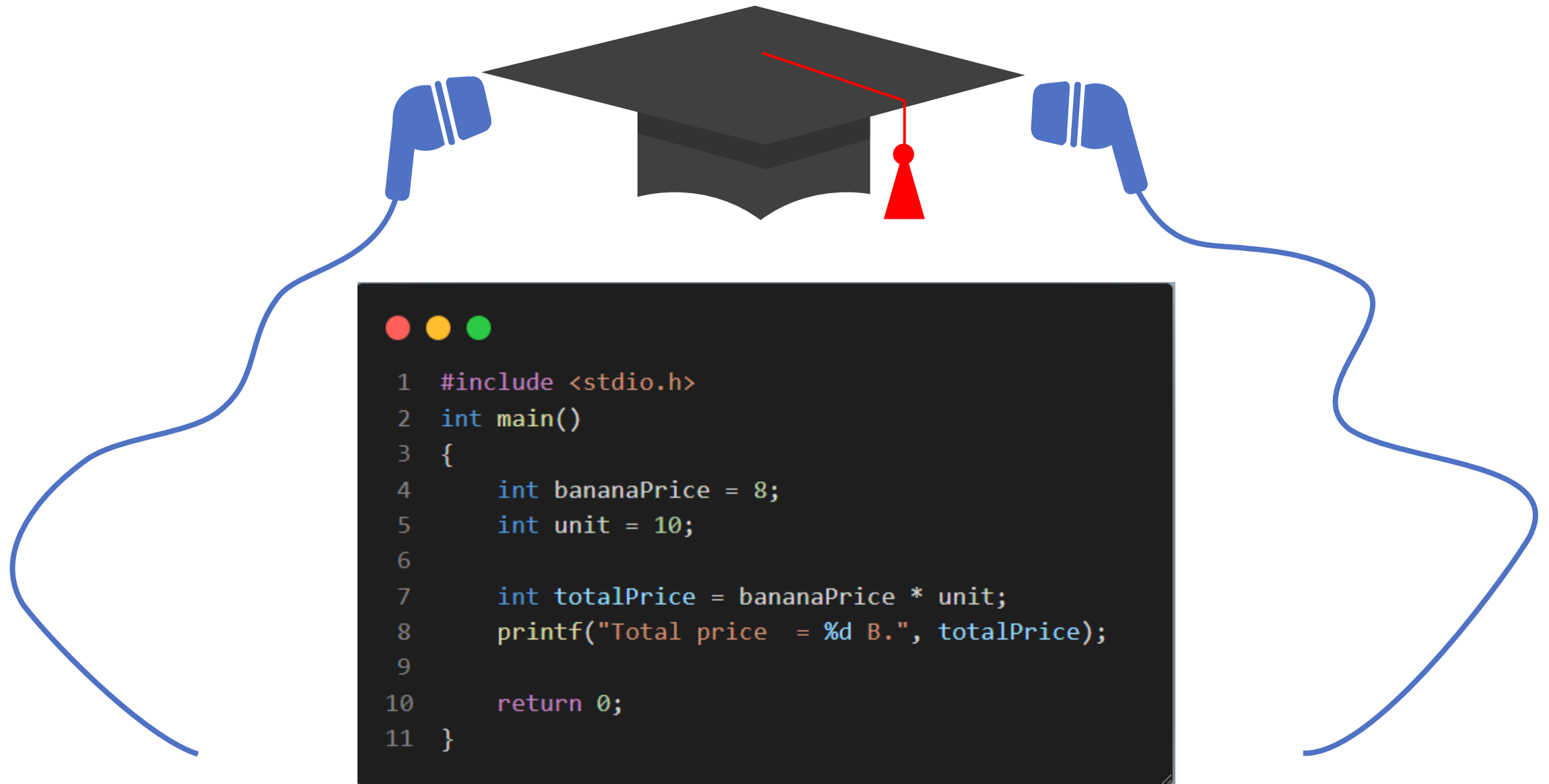
```
1 #include <stdio.h>
2 int main()
3 {
4     char name[3] = "sad";
5     return 0;
6 }
```

s	a	d
0	1	2

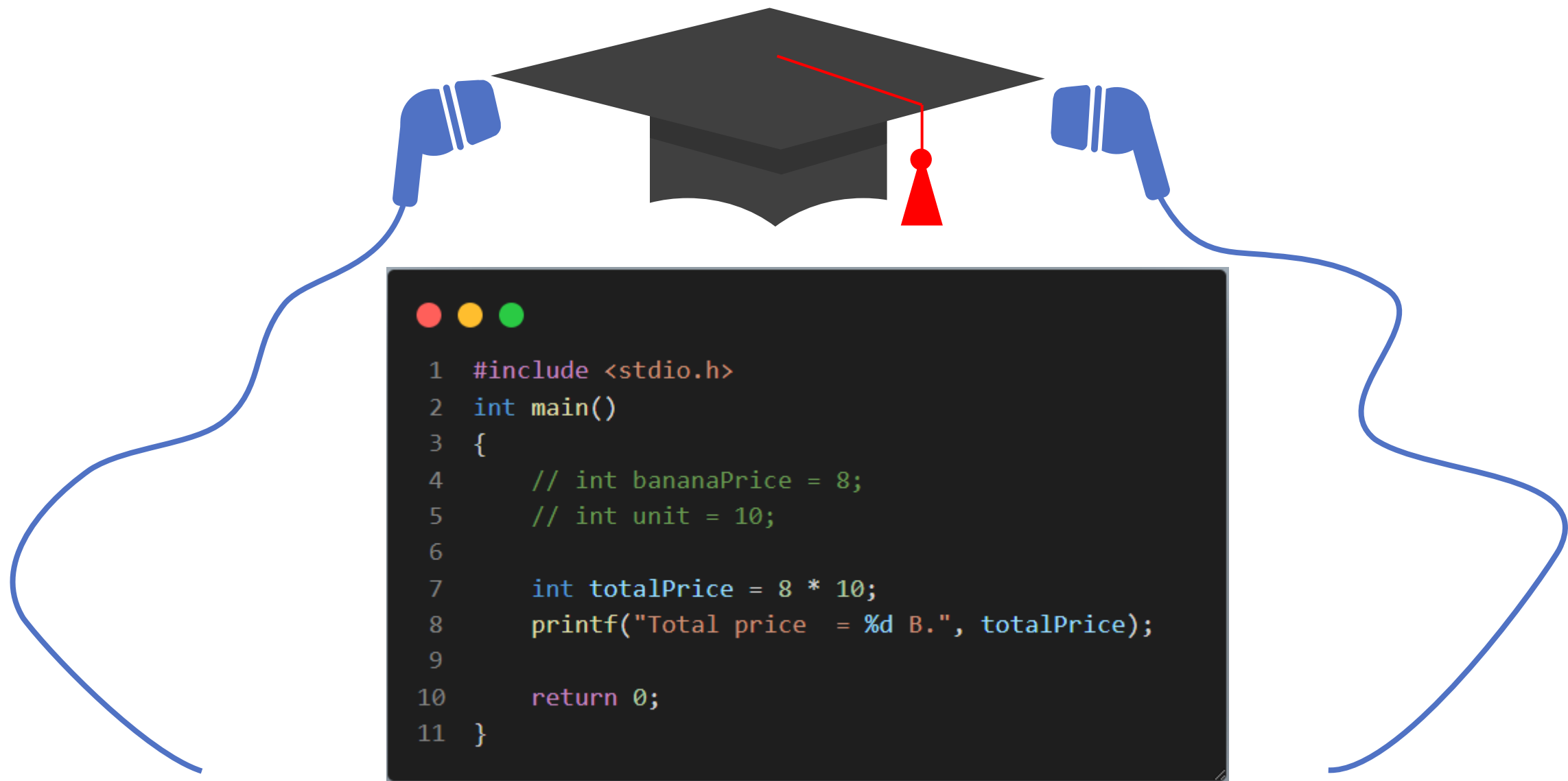


ประโยชน์
ของตัวแปร

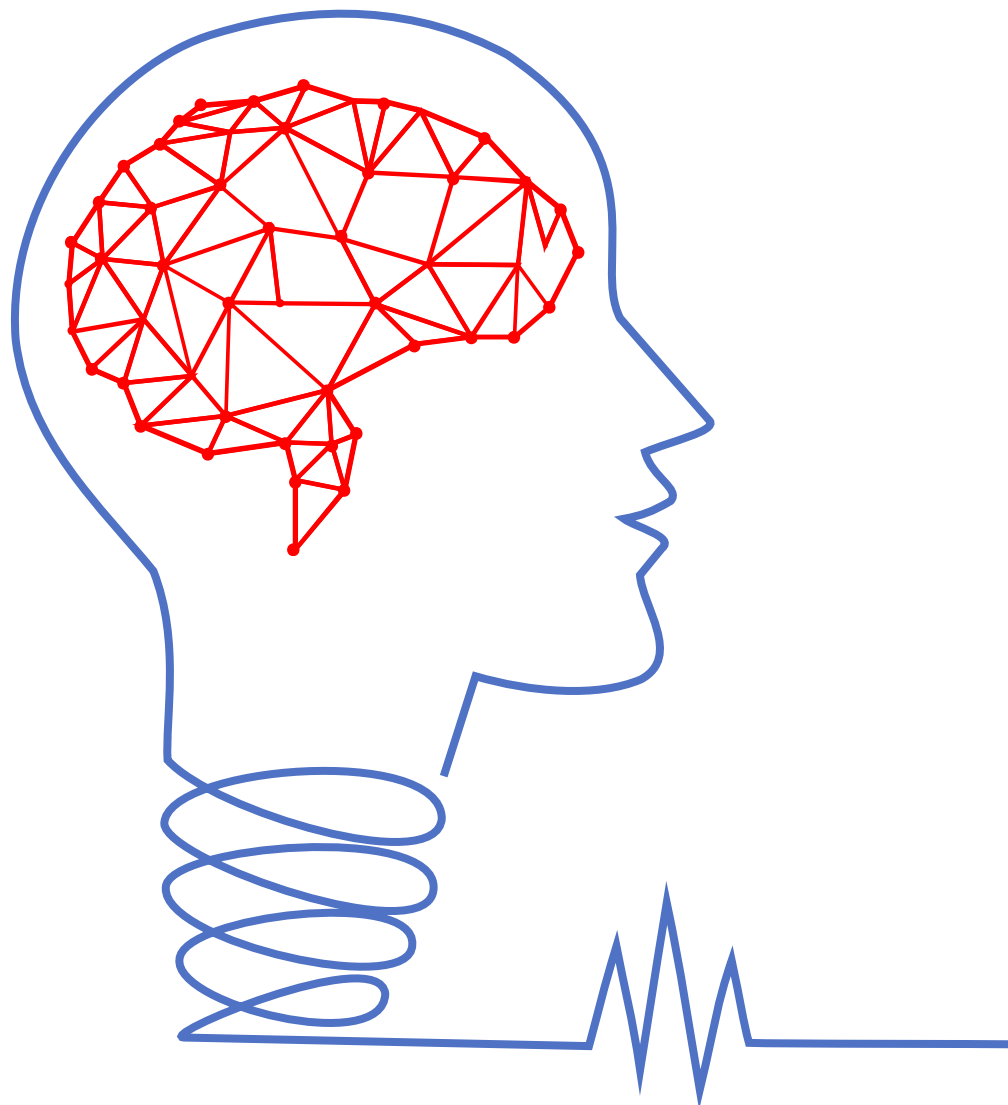
- ทำให้ Code เข้าใจง่ายขึ้น
- สามารถใช้ซ้ำๆ ได้ในหลายๆ ที่
- แก้ไขค่าตรงไหนก็ได้



```
1  #include <stdio.h>
2  int main()
3  {
4      int bananaPrice = 8;
5      int unit = 10;
6
7      int totalPrice = bananaPrice * unit;
8      printf("Total price = %d B.", totalPrice);
9
10     return 0;
11 }
```



Variable Naming Rules



01

ต้องไม่มีอักขรพิเศษใดๆ ประกอบอยู่ด้วย เช่น ! @ # \$ % ^ & * (

02

สามารถใช้เครื่องหมาย underscore (_)

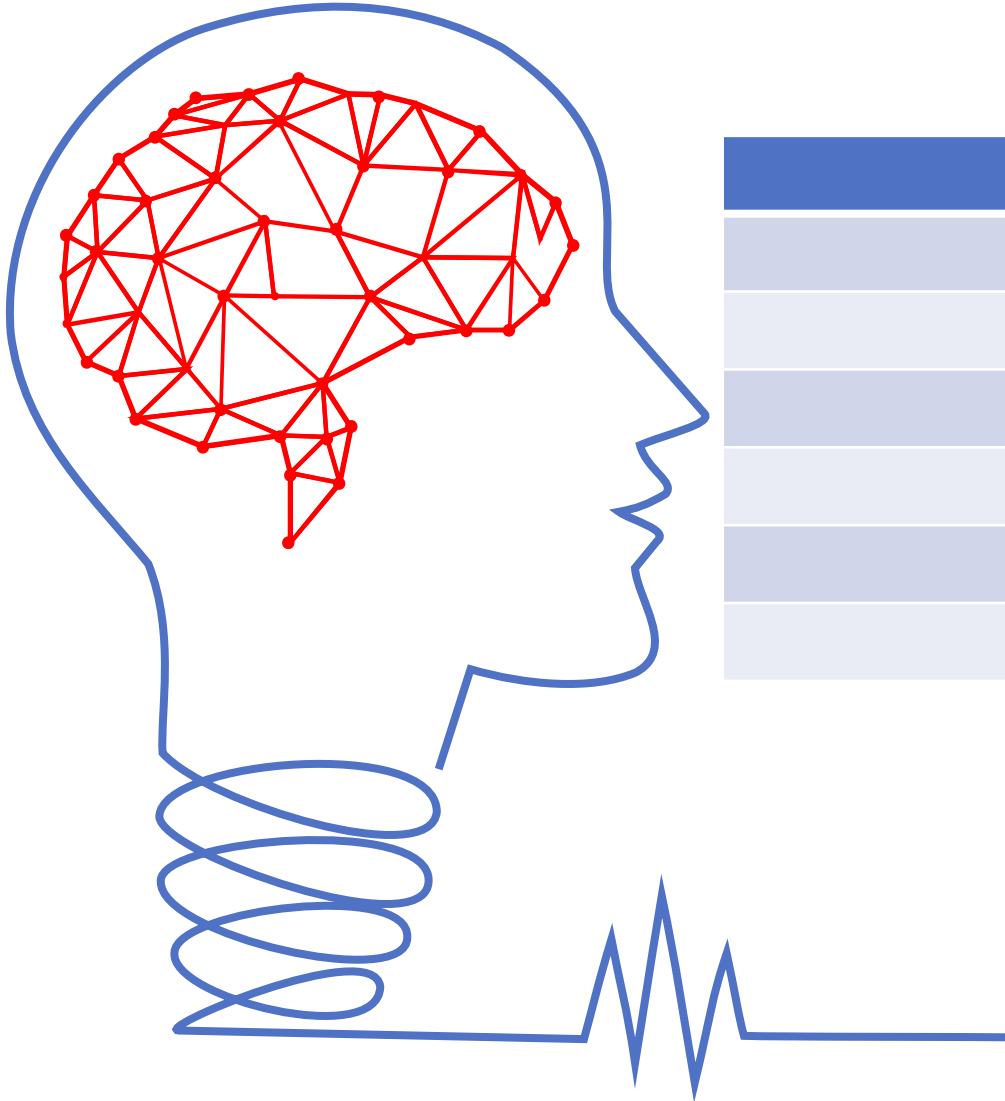
03

ชื่อตัวแปรต้องมีตัวเลขปนอยู่ได้ แต่ต้องไม่ขึ้นต้นด้วยตัวเลข
ห้ามมีช่องว่างระหว่างชื่อ

04

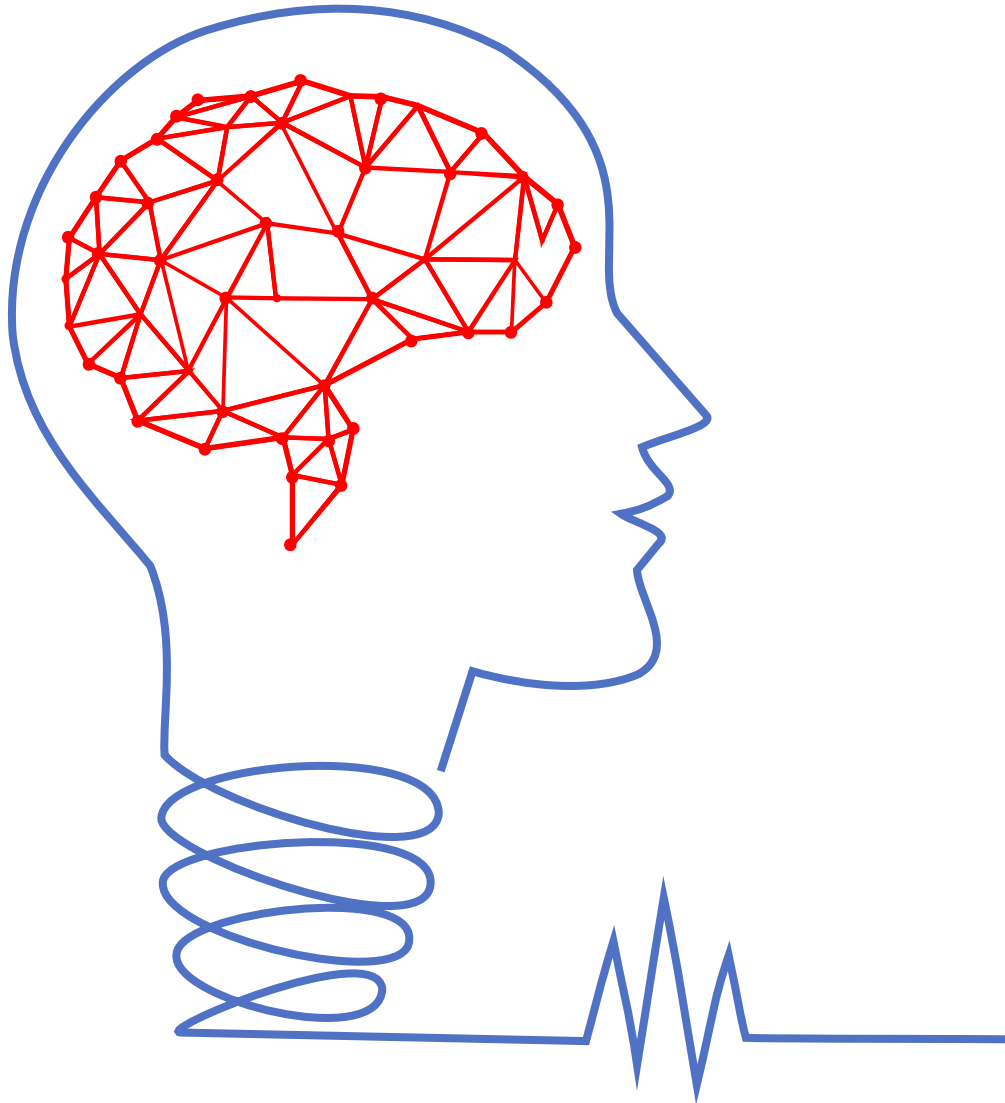
ใช้ได้ทั้งพิมพ์เล็ก และพิมพ์ใหญ่
ชื่อเหมือนกันแต่เป็นพิมพ์เล็กพิมพ์ใหญ่ ถือว่าคนละชื่อกัน
ห้ามตั้งชื่อซ้ำกับคำสงวน เช่น char long while do

Displaying values from variables



สัญลักษณ์	ใช้แบบ
%d	เลขจำนวนเต็ม
%s	สตริง
%f	เลขทศนิยม
%c	ตัวอักษร 1 ตัว
%o	ตัวเลขในฐาน 8
%x	ตัวเลขในฐาน 16

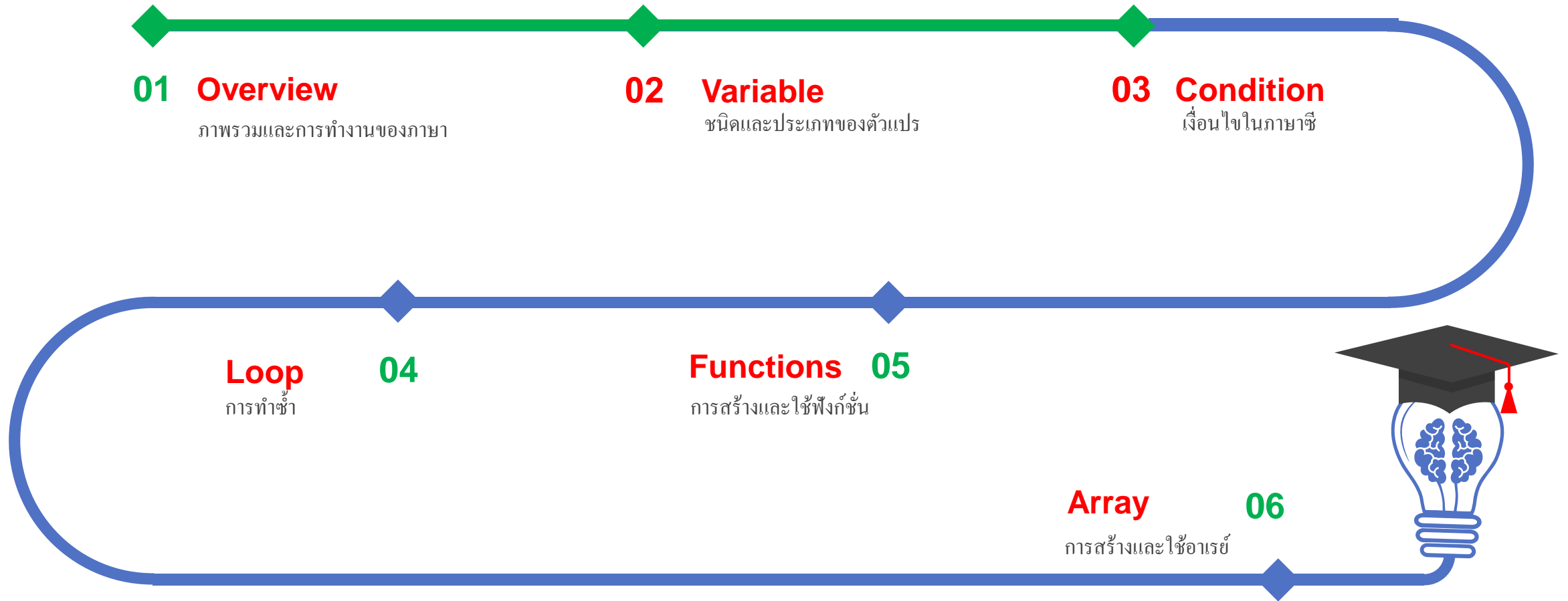
Constants in C language

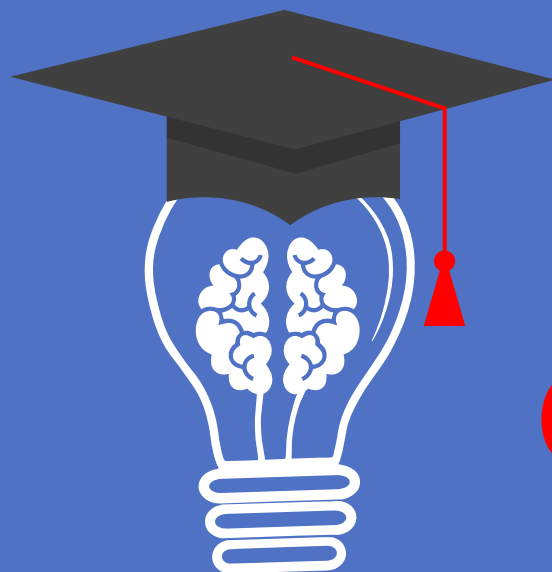


01 `const`

02 `#define`

Timeline Style





Condition

เงื่อนไข

Comparison mark

<, >

มากกว่า น้อยกว่า

>=, <=

มากกว่าเท่ากับหรือไม่ น้อยกว่าเท่ากับหรือไม่

!=

ไม่เท่ากันหรือไม่

==

เท่ากันใช่หรือไม่



- && คือ เครื่องหมาย “และ” (AND)
- || คือ เครื่องหมาย “หรือ” (OR)



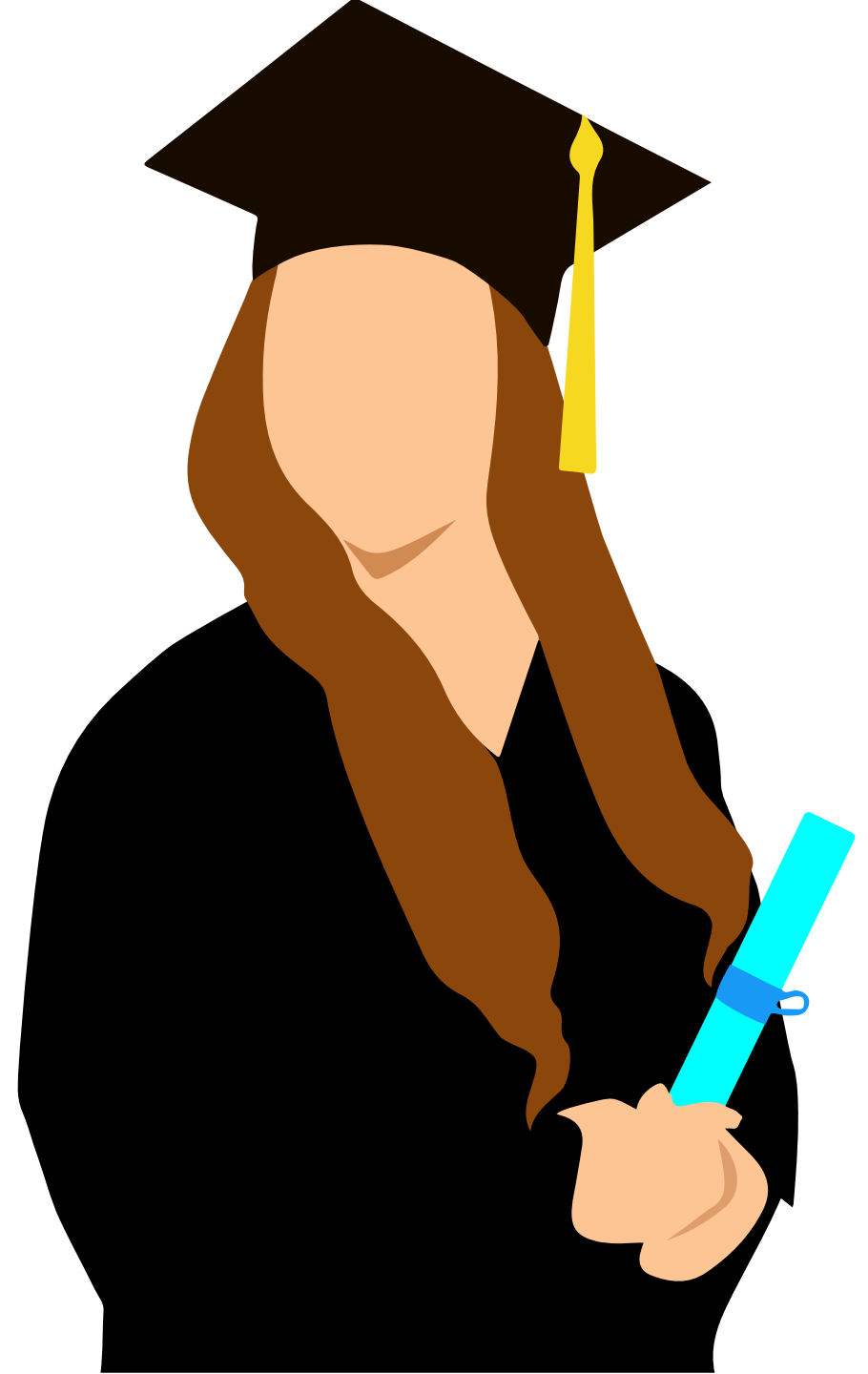
If

คำสั่งที่ใช้เปรียบเทียบว่า เป็น จริง หรือ เท็จ

รูปแบบการเขียน

```
if (เงื่อนไขการเปรียบเทียบ) {  
    สิ่งที่ต้องการให้ทำเมื่อเงื่อนไขเป็นจริง  
}
```

```
1  #include <stdio.h>  
2  int main()  
3  {  
4      int age;  
5      printf("How old are you : ");  
6      scanf("%d", &age);  
7      if (age >= 60)  
8      {  
9          printf("Old ");  
10     }  
11     return 0;  
12 }
```



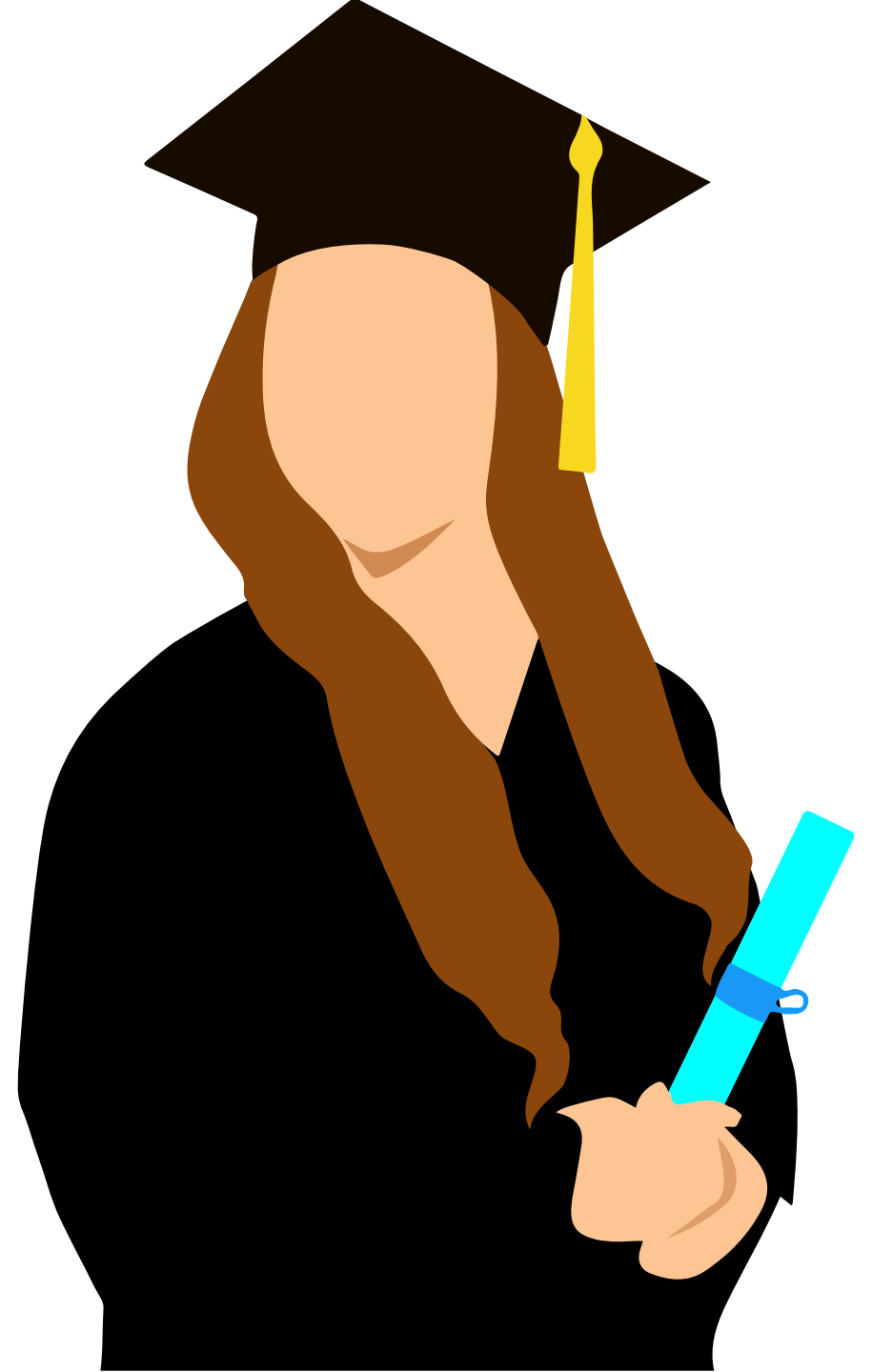
else

เปรียบเทียบ 2 เงื่อนไข และเป็นตัวจบของ if

รูปแบบการเขียน

```
else {  
    สิ่งที่ต้องการให้ทำเมื่อเงื่อนไขเป็นจริง  
}
```

```
1  int main()  
2  {  
3      int age;  
4      printf("How old are you : ");  
5      scanf("%d", &age);  
6      if (age >= 60)  
7      {  
8          printf("Old ");  
9      }  
10     else  
11     {  
12         printf("Young");  
13     }  
14     return 0;  
15 }
```



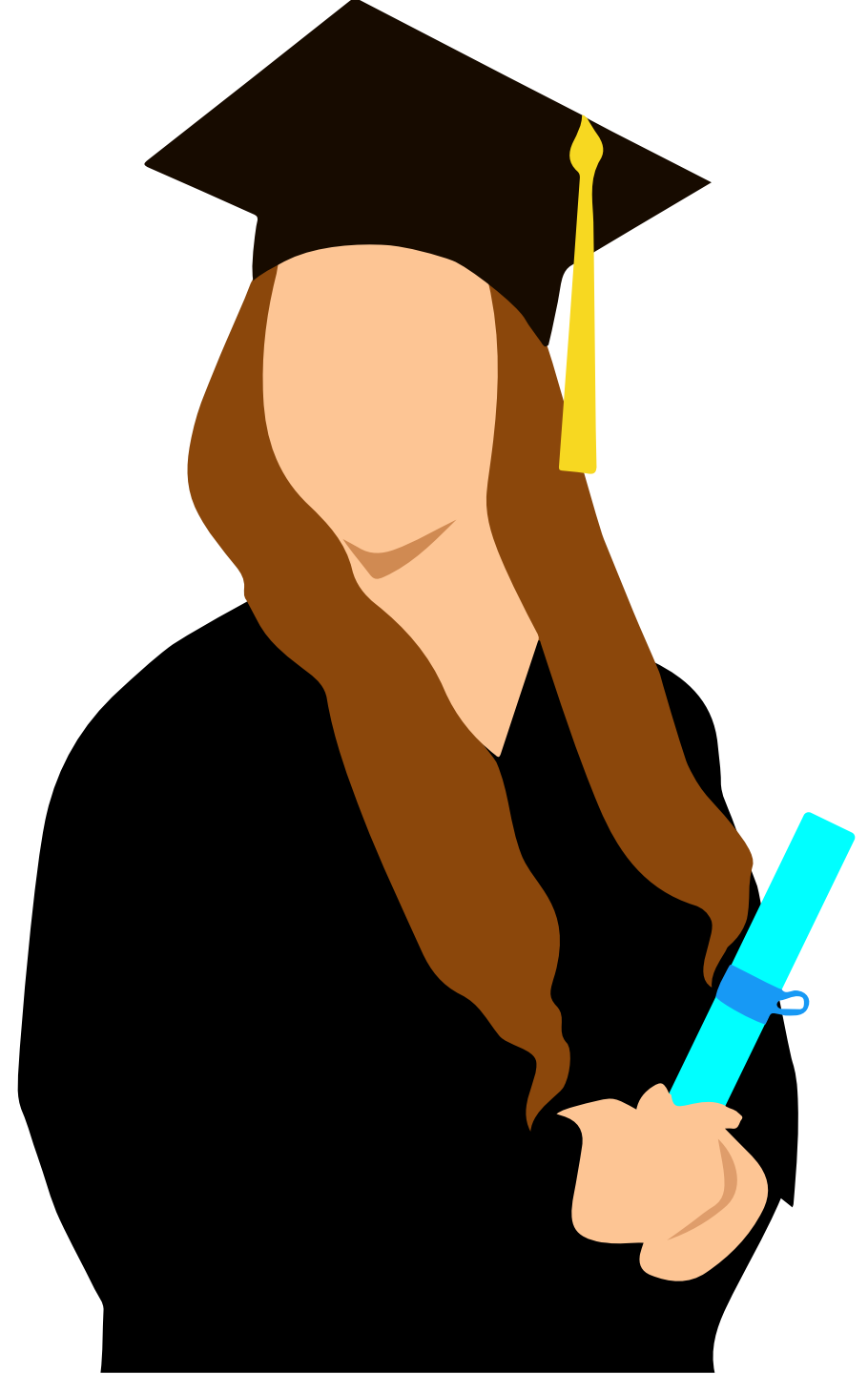
else If

เปรียบเทียบ 3 เงื่อนไขขึ้นไป และเป็นตัวต่อของ if

รูปแบบการเขียน

```
else if (เงื่อนไขการเปรียบเทียบ) {  
    สิ่งที่ต้องการให้ทำเมื่อเงื่อนไขเป็นจริง  
}
```

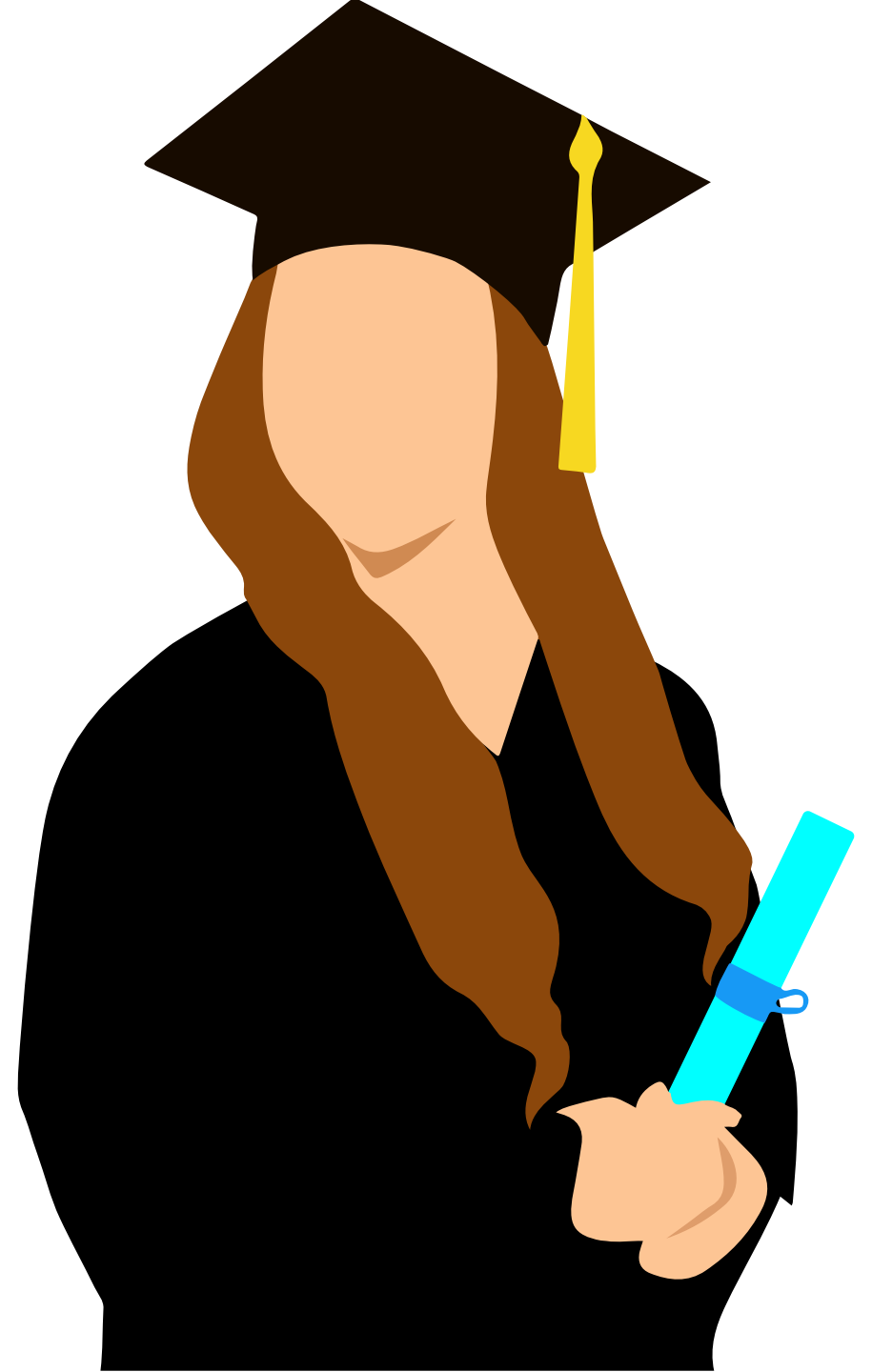
```
1  #include <stdio.h>  
2  int main()  
3  {  
4      int age;  
5      printf("How old are you : ");  
6      scanf("%d", &age);  
7      if (age >= 60)  
8      {  
9          printf("Old ");  
10     }  
11     else if (age >= 50 && age <= 59)  
12     {  
13         printf("Middle man");  
14     }
```



Switch..case

รูปแบบการเขียน

```
switch (ตัวแปรที่ใช้เช็คค่า)
{
    case `` : สิ่งที่ต้องการให้ทำเมื่อเงื่อนไขเป็นจริง
    case `` : สิ่งที่ต้องการให้ทำเมื่อเงื่อนไขเป็นจริง
    default `` : สิ่งที่ทำเพราะไม่ตรงเงื่อนไขใดๆเลย
}
```



Proposition

หาพื้นที่วงกลม

โดยใช้ความรู้เรื่องค่าคงที่ในการแก้ปัญห

คำนวณราคาผลไม้

รับค่าจำนวนทุเรียนมาแล้วให้หาว่าถ้าซื้อ
เท่ากับจำนวนที่รับค่ามาจะเป็นเงินทั้งหมด
เท่าไร โดยราคาต่อลูกคือ 180

โปรแกรมตัดเกรด

โดยใช้ความรู้เรื่อง and or เข้ามาแก้ปัญห



แสดงผลตามภาพ

รับค่าเข้า มา 3 แสดง

*
* *
* * *

รับได้มากที่สุด 10 ค่า

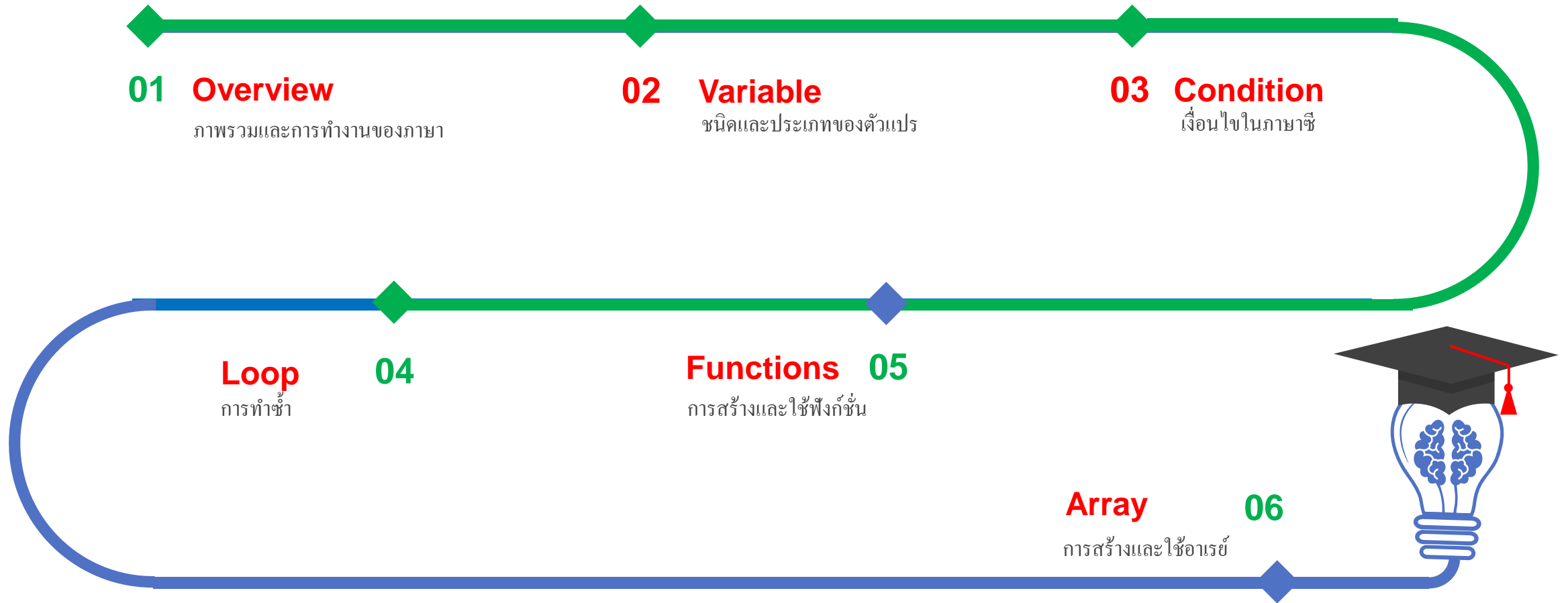
ใช้เราไหมนะ

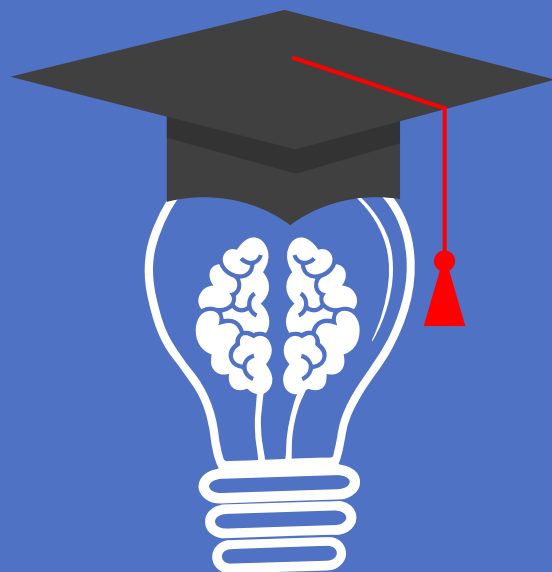
รับค่าตัวย่อชื่อจริงของเรา
ถ้าใช้ให้ แสดงผลออกมา ชื่อจริงเรา
อายุเรา เพศ
และสิ่งที่สนใจเกี่ยวกับภาษาทาง
คอมพิวเตอร์อย่างน้อย 2 ภาษา

ช่วยฉันคำนวณหน่อย

ถ้ารับค่า 1-10 ให้เอาค่าไป คูณกับ 10
แต่ถ้ารับค่า อื่นให้เอาหารด้วย 10
แสดงผลเป็นค่าของคำตอบ

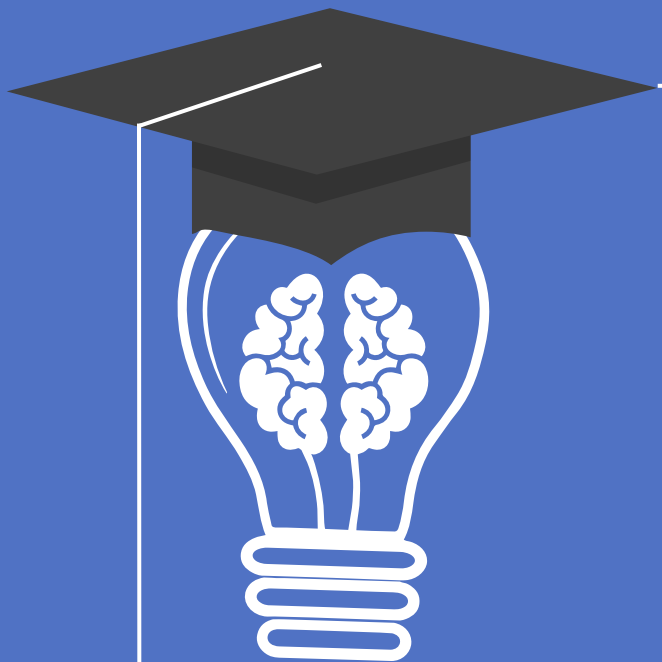
Timeline Style





Loop
การทำงานซ้ำ

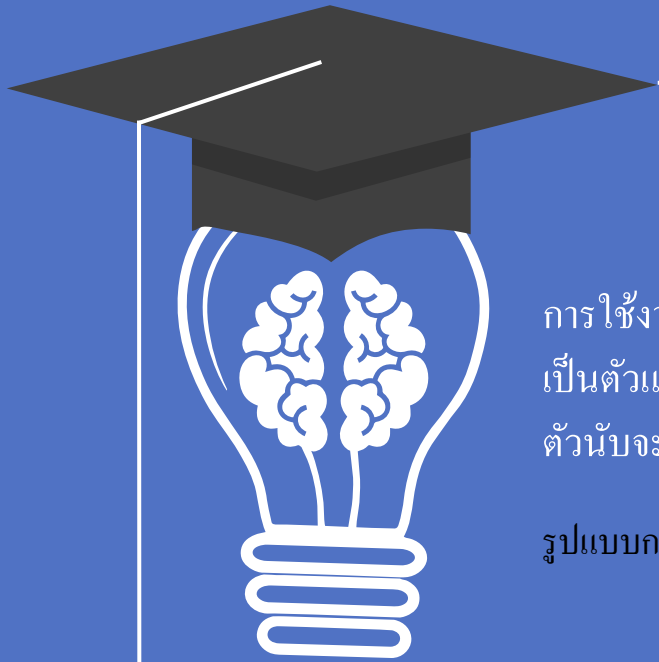
Loop



การทำงานซ้ำมี 2 ลักษณะ

- คือ 1. การทำงานที่รู้จำนวนรอบการทำงานที่แน่นอนว่าต้องทำกี่รอบ (for)
2. การทำงานที่ต้องเช็คตลอดว่าตรงเงื่อนไขไหม(while , do while)

For Loop



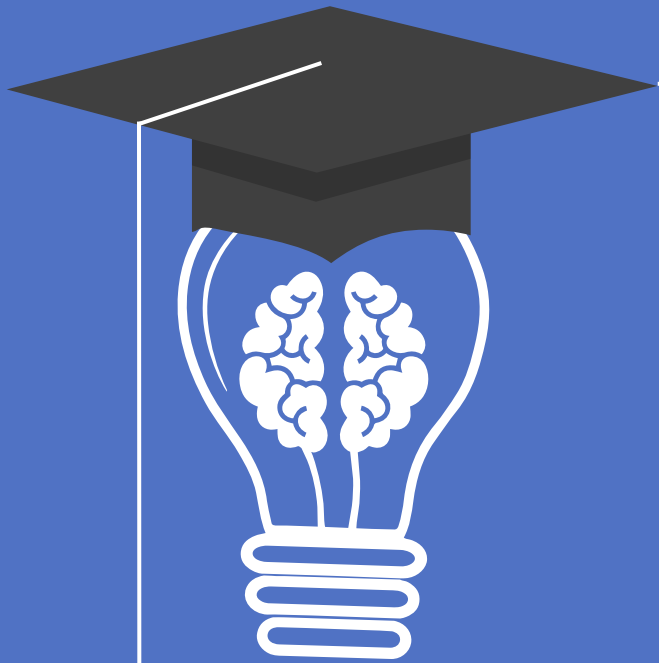
การใช้งานจะกำหนดลงไปเลยว่าต้องการให้ทำกี่ครั้ง และจะต้องสร้างตัวแปร ขึ้นมาเป็นตัวนับ (counter) โดยมากจะกำหนดให้เป็นตัวแปรแบบจำนวนเต็ม (integer)

ตัวนับจะเป็นตัวคอยบอกว่าครบตามจำนวนที่ต้องการรึยัง ถ้ายังไม่ครบก็จะวนไปเรื่อยๆ จนกว่าจะครบ

รูปแบบการเขียน

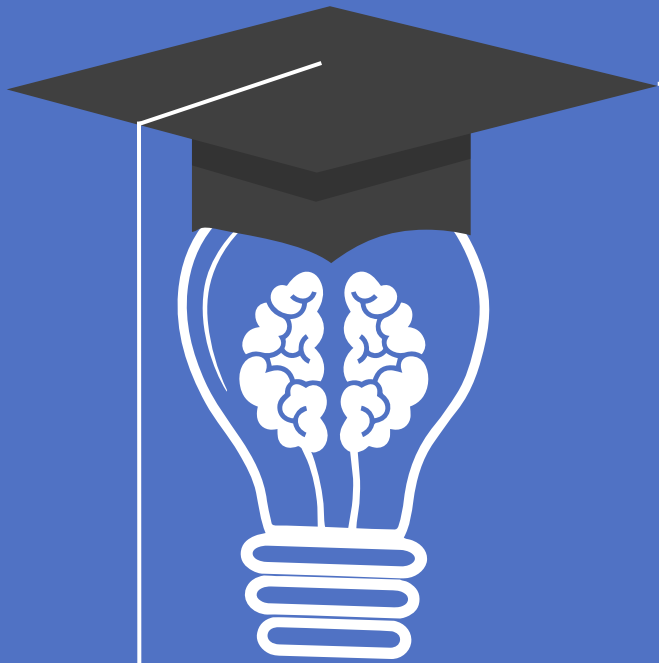
```
for (ค่าเริ่มต้นของตัวนับ เช่น 0 ;เปรียบเทียบตัวนับว่ายังอยู่ในช่วงนี้หรือไม่;จำนวนนับเพิ่มทีละเท่าไร หรือลดทีละเท่าไร){  
    สิ่งที่ต้องการให้ทำ  
}
```

For Loop



```
1  #include <stdio.h>
2  int main()
3  {
4      /* code */
5      int count = 10;
6      for (int i = 0; i < count; i++)
7      {
8          printf("รอบที่ %d Hi!\n", i);
9      }
10
11     return 0;
12 }
```

While Loop



```
1  int count = 1;
2  while (count <= 10)
3  {
4      printf("Hello % d \n", count);
5      count++;
6  }
7
```

จากตัวอย่างเรากำหนดให้ count มีค่าเป็น 1 จากนั้นใช้คำสั่ง while ตรวจสอบค่าของ count ว่ายังน้อยกว่าหรือเท่ากับ 10 หรือไม่ ถ้ายังเป็นจริงจะทำคำสั่งภายในปีกกา {} ต่อไป และเพิ่มค่า count เพิ่มทีละ 1 ไปจนกว่าค่าของ count จะมากกว่า 10 มันจะทำให้เงื่อนไขเป็นเท็จก็จะหลุดจากลูป while

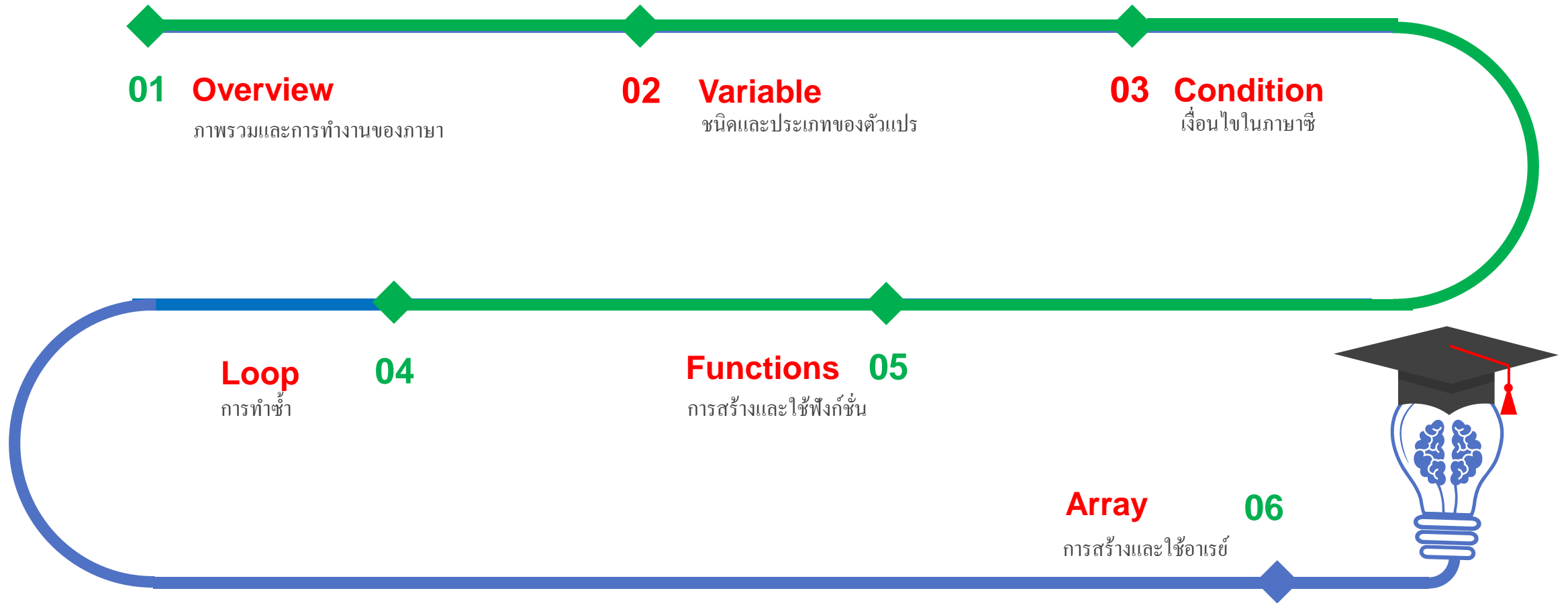
Do While Loop

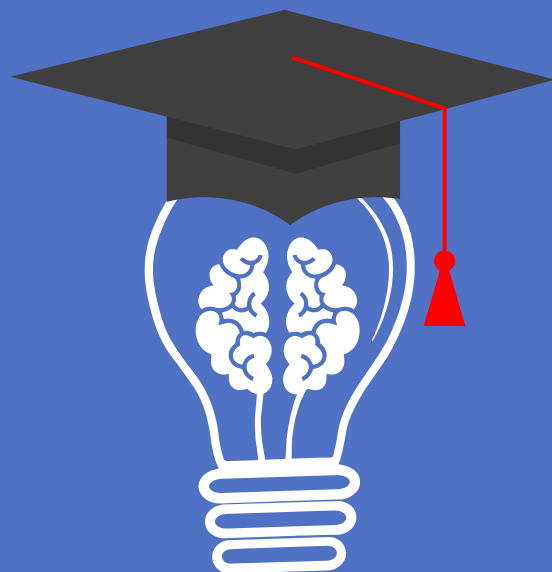


```
1  int n = 1;
2      do
3      {
4          printf("%d\n", n);
5          n++;
6      } while (n <= 10);
```

ก่อนเข้าสู่การตรวจสอบเงื่อนไข ($n \leq 10$) ครั้งแรกนั้นจะมีการทำคำสั่งใน `{}` ก่อน ดังนั้น `do..while` จึงทำคำสั่งภายใน `{}` อย่างน้อย 1 ครั้งเสมอ และถ้าตรวจสอบแล้วเงื่อนไขยังจริงอยู่ก็จะวนมาทำคำสั่งภายใน `{}` อีกครั้ง

Timeline Style





Function

การสร้างและใช้งาน

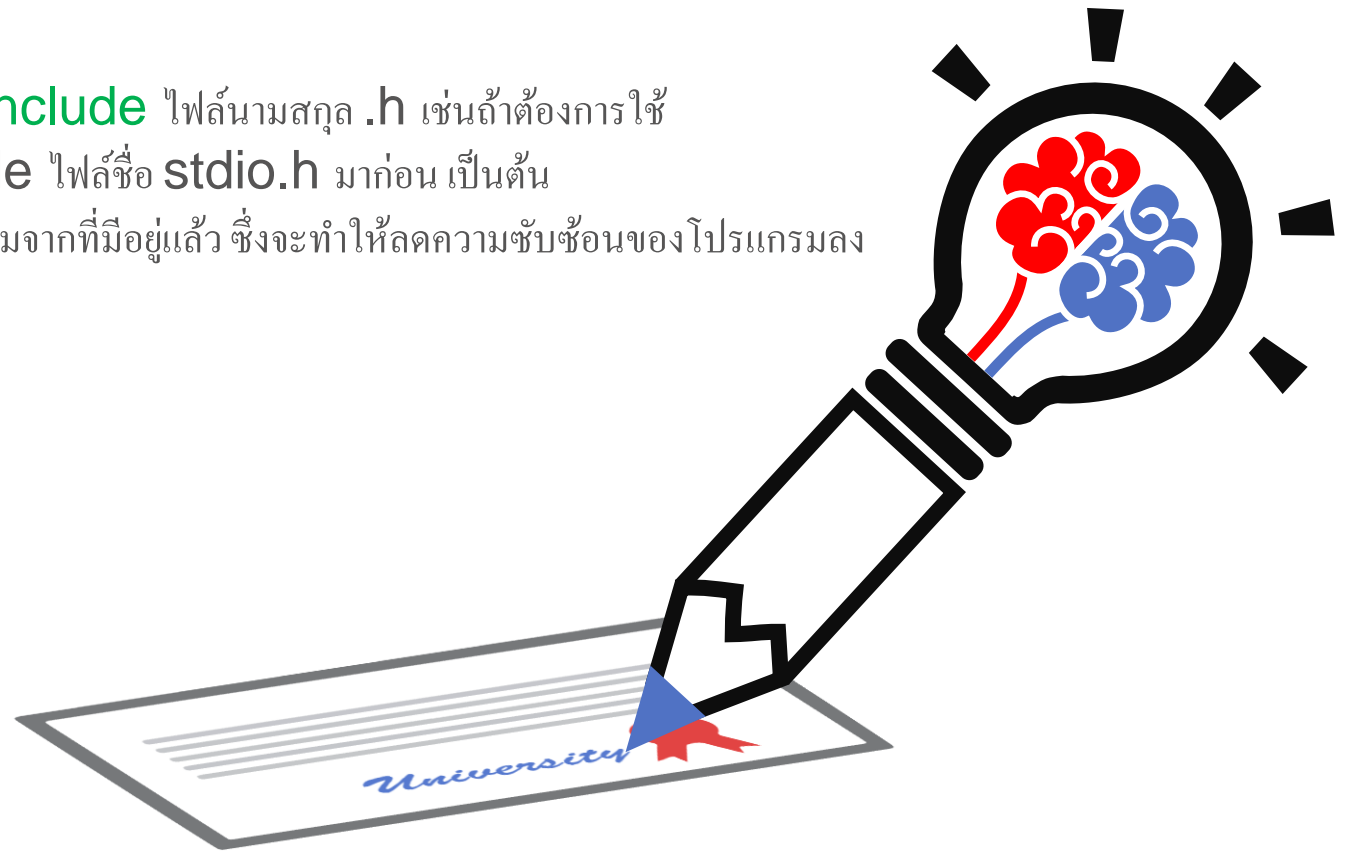
ฟังก์ชันในภาษาซีมีอยู่ 2 ประเภทคือ

- **Function** ที่มีอยู่ในไลบรารีของภาษาซี
- **Function** ที่เราสร้างขึ้นเอง

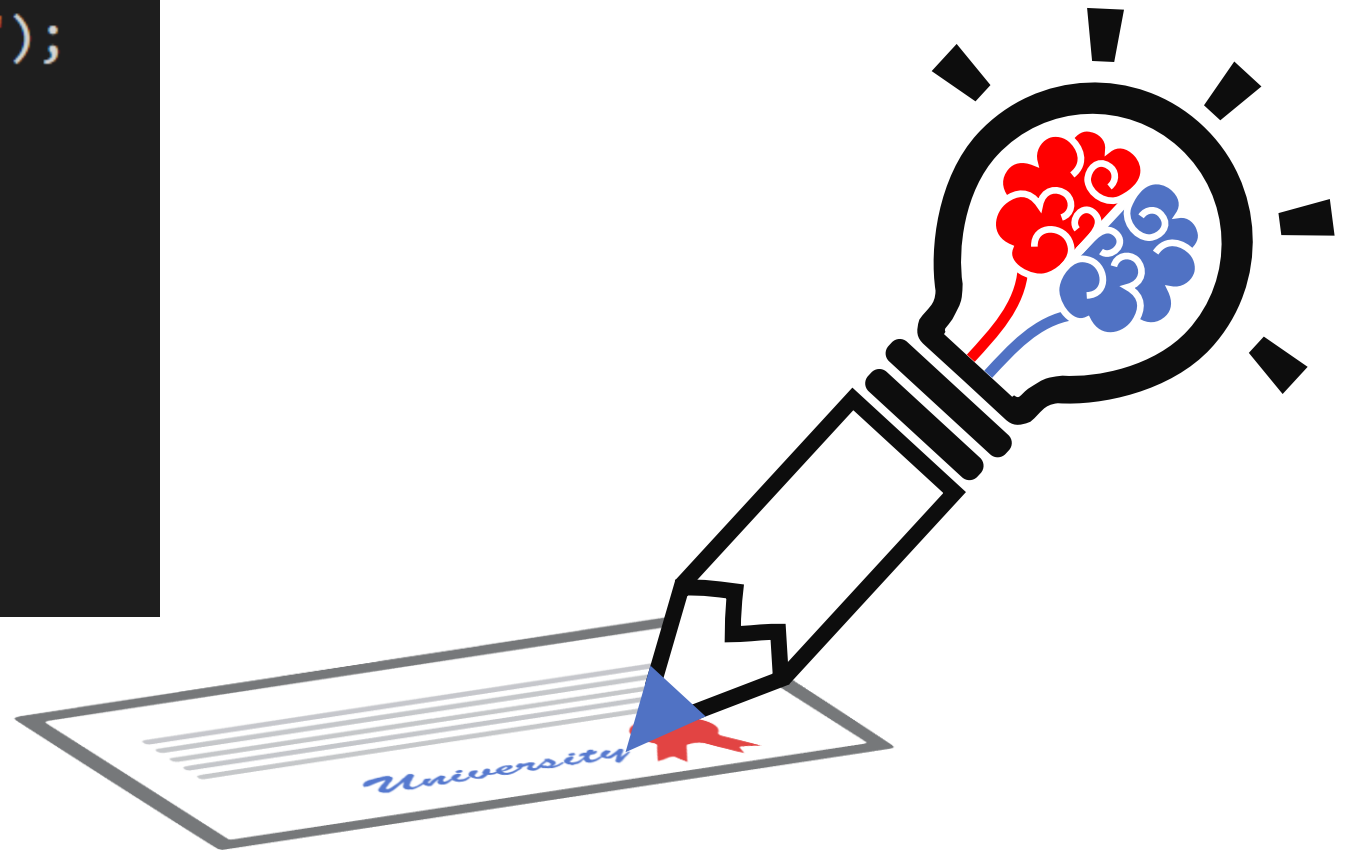
Function ที่มีอยู่แล้วในภาษาซีเวลาจะเรียกใช้งานก็ต้องทำการ **include** ไฟล์นามสกุล **.h** เช่นถ้าต้องการใช้

Function ไม่ว่าจะเป็น **printf** หรือ **scanf** จะต้อง **include** ไฟล์ชื่อ **stdio.h** มาก่อน เป็นต้น

Function ที่เขียนขึ้นเองใหม่เหมือนกับสร้างภาษาซีขึ้นเองเพิ่มเติมจากที่มีอยู่แล้ว ซึ่งจะช่วยให้ลดความซับซ้อนของโปรแกรมลง



```
1  #include <stdio.h>
2  void test()
3  {
4      printf("Hi");
5  }
6  int main()
7  {
8      /* code */
9      test();
10     return 0;
11 }
```



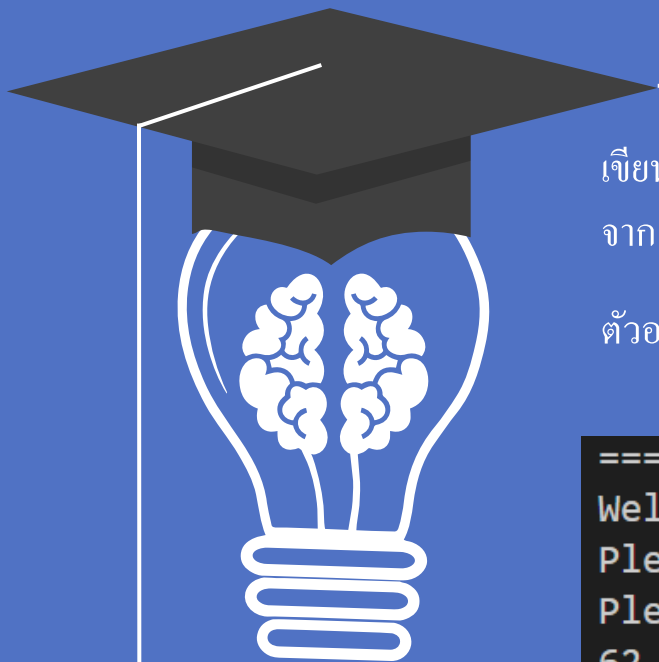


เขียนโปรแกรม เพื่อคำนวณหาพื้นที่ของสามเหลี่ยม จากสูตร $\text{Area} = 1/2 \times \text{ฐาน} \times \text{สูง}$ รับค่า (Input) คือ Base และ Hight (Hint: ต้องใช้ #define)

ตัวอย่างการทำงาน

```
=====
Welcome, this is a program for calculating the area of a triangle.
Please enter the value of the base of the triangle : 28
Please enter the height of the triangle : 37
518.00
=====
```

```
=====
Welcome, this is a program for calculating the area of a triangle.
Please enter the value of the base of the triangle : 18.00
Please enter the height of the triangle : 18.00
162.00
=====
```



เขียนโปรแกรม เพื่อคำนวณหาปริมาตรของรูปทรงกระบอกจากสูตร $\text{Volume} = \text{Area} \times H$ และพื้นที่วงกลม
จาก $\text{Area} = \pi r(2)$ รับค่า (Input) คือ Radius และ Height

ตัวอย่างการทำงาน

```
=====
Welcome, this is a program for calculating the find the volume of a cylinder.
Please enter the radius of the sphere : 2
Please enter the height of the cylindrical : 5
62.80
=====
```

```
=====
Welcome, this is a program for calculating the find the volume of a cylinder.
Please enter the radius of the sphere : 5
Please enter the height of the cylindrical : 2
157.00
=====
```



เขียนโปรแกรม เพื่อรับค่าของเวลา ณ ปัจจุบัน ข้อมูลรับเข้าตามตัวอย่าง

ตัวอย่างการทำงาน

```
Enter current Hour = 22  
Enter current Min = 22  
Enter current Sec = 22  
Current time is 22:22:22
```

```
Enter current Hour = 26  
Error, enter within 0-24 range.
```



เขียนโปรแกรม เพื่อรับค่าของผลสอบ จากสูตร $\text{Total} = \text{Midterm} + \text{Final} + \text{HW}$ คะแนนแบ่งเป็น 30 50 20 ตามลำดับ

ตัวอย่างการทำงาน

```
Enter final exam scores = 49
Enter midterm exam scores = 20
Enter homework scores= 10
Total = 79
```

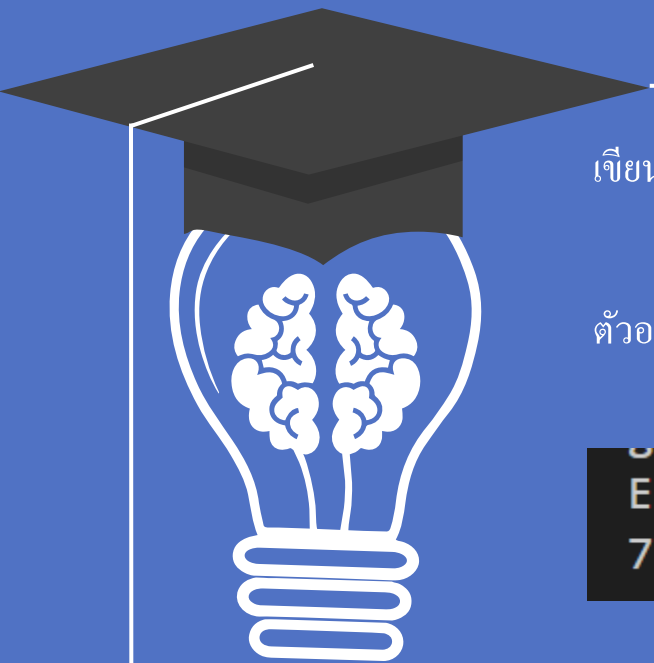
```
Enter final exam scores = 90
Error, enter within 0-50 range.
```



เขียนโปรแกรมคำนวณ Mean รับ input 5 ตัว

ตัวอย่างการทำงาน

```
Enter the number you want to find the mean digit 1 : 2.0
Enter the number you want to find the mean digit 2 : 3.0
Enter the number you want to find the mean digit 3 : 2.0
Enter the number you want to find the mean digit 4 : 3.0
Enter the number you want to find the mean digit 5 : 2.0
Mean : 2.40
```

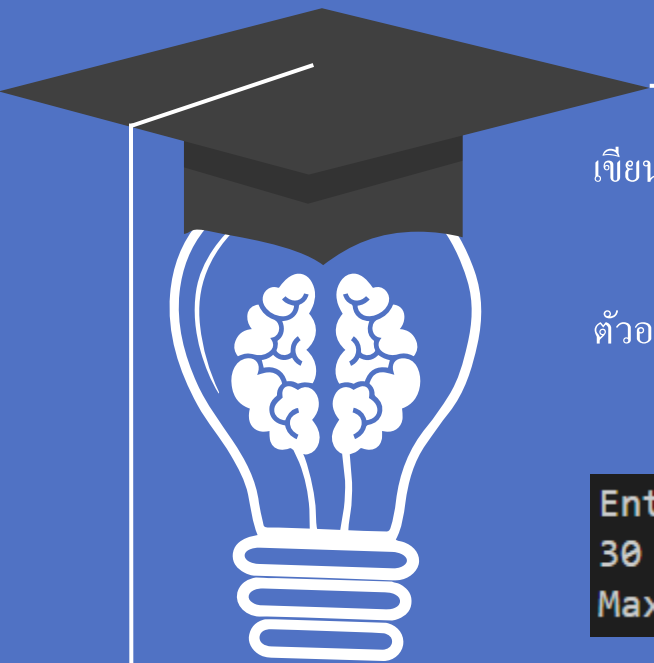



เขียนโปรแกรมรับค่าของตัวเลข 1 ตัวคือ (x) แล้วทดสอบว่า เป็นเลขที่หารด้วย 7 ลงตัวหรือไม่

ตัวอย่างการทำงาน

```
Enter x :7  
7 is divisible by 7
```

```
Enter x :9  
Not divisible by 7
```



เขียนโปรแกรมรับค่าของตัวเลข 2 ค่า (x, y) และทดสอบเงื่อนไขว่าค่าใดมากที่สุด

ตัวอย่างการทำงาน

```
Enter number 1 or 2 :10  
30  
Maximum is : 30
```

```
Enter number 1 or 2 :25  
49  
Maximum is : 49
```

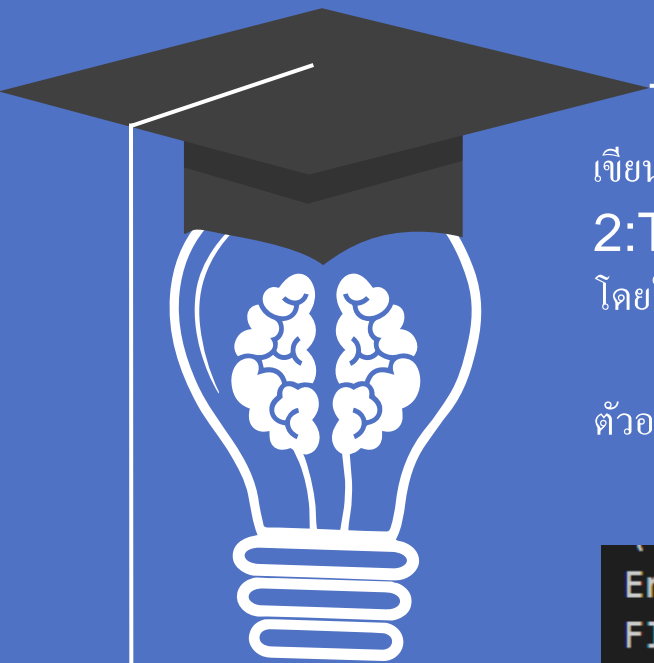


โปรแกรมรับค่าของตัวเลือก 1 ค่า (x) และทดสอบว่า เป็นการหาพื้นที่สี่เหลี่ยม (Rectangle Area) หรือสามเหลี่ยม (Triangle Area) จากนั้นรับค่าตัวแปรเพื่อหาพื้นที่ Hint ใช้ switch case

ตัวอย่างการทำงาน

```
emrest\rectangle_triangle_area  
Select 1.(Rectangle) or 2.(Triangle) :2  
Enter base : 10  
Enter hight : 20  
Triangle Area = 200
```

```
Select 1.(Rectangle) or 2.(Triangle) :1  
Enter width : 10  
Enter length : 20  
Rectangle Area = 100
```



เขียนโปรแกรมรับค่าของเลข (N) และแสดงตัวเลขตัวสุดท้ายของ N เป็นดังนี้(เช่น 0:Zero, 1:One, 2:Two, 3:Three, 4:Four, 5:Five, 6:Six, 7:Seven, 8:Eight, 9:Nine)
โดยใช้คำสั่ง switch

ตัวอย่างการทำงาน

```
Enter an integer N = 155  
FIVE
```

```
Enter an integer N = 999  
NINE
```

Category	Operator	Associativity
Postfix	() [] -> . ++ --	Left to right
Unary	+ - ! ~ ++ -- (type) * & sizeof	Right to left
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Shift	<< >>	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Bitwise AND	&	Left to right
Bitwise XOR	^	Left to right
Bitwise OR		Left to right
Logical AND	&&	Left to right
Logical OR		Left to right
Conditional	?:	Right to left
Assignment	= += -= *= /= %= >>= <<= &= ^= =	Right to left
Comma	,	Left to right