Microcontrollers & Applications

Lecture 4.1: Operators

Operator Presedence (C & Python)

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

Operators	Meaning
`()`	Parentheses
`**`	Exponent
`+x`, `-x`, `~x`	Unary plus, Unary minus, Bitwise NOT
`*`, `/`, `//`, `%`	Multiplication, Division, Floor division, Modulus
`+`, `-`	Addition, Subtraction
`<<`, `>>`	Bitwise shift operators
`&`	Bitwise AND
`^`	Bitwise XOR
`|`	Bitwise OR
`==`, `!=`, `>`, `>=`, `<`, `<=`, `is`, `is not`, `in`, `not in`	Comparisons, Identity, Membership operators
`not`	Logical NOT
`and`	Logical AND
`or`	Logical OR

Comparison Operators (C & Python)

- x equals to 5 — x == 5
- x is different than 5 — x != 5
- x less than 5 — x < 5
- x greater than 5 — x > 5
- x less than or equals to 5 — x <= 5
- x greater than or equals to 5 — x >= 5

Logic Operators (C & Python)

- «not» operator runs first, then other operators run from left to right.
- If both condition#1 and condition#2 are true

  condition#1 && condition#2

- If either condition#1 or condition#2 are true

  condition#1 || condition#2

- If condition#1 is NOT true

  ! condition#1

- «not» operator runs first, then other operators run from left to right.
- If both condition#1 and condition#2 are true

  condition#1 and condition#2

- If either condition#1 or condition#2 are true

  condition#1 or condition#2

- If condition#1 is NOT true

  not condition#1

Arithmetic Operators (C & Python)

- Increment one: x++ ++x
- Decrement one: x-- --x
- Addition: x+5
- Subtraction: x-5
- Multiplication: x*5
- Division: x/5
- Modulus: x%5
- Exponent: pow(x,5)
- Integer division: x/5 with int x definition

- Increment one: x=x+1
- Decrement one: x=x-1
- Addition: x+5
- Subtraction: x-5
- Multiplication: x*5
- Division: x/5
- Modulus: x%5
- Exponent: x**5
- Integer division: x//5

Bitwise Operators (C & Python)

- Bitwise and: &
- Bitwise or: |
- Bitwise not: ~
- Bitwise xor: ^

Assignment Operators (C & Python)

- Assignment: x=5
- Augmented addition: x+=5 → x=x+5
- Augmented subtraction: x-=5 → x=x-5
- Augmented multiplication: x*=5 → x=x*5
- Augmented division: x/=5 → x=x/5
- Augmented modulus: x%=5 → x=x%5
- Augmented floor division: x//=5 → x=x//5 (Python only)

Left to Students (C)

- What would be the value of 'a’:
  - int a = 10/45*23%45/(45%4*21)
  - float a = 10+45.0*23-45+(4*21.0)
- True or false:
  - 4>5 && 5>4
  - 4>5 || 5>4
  - (232+23*1233) || 0
  - (232+23 *1233) && 0
- What would be the output of:
  - Serial.println(1==5==5);
- If a is 15, then what would be screened and the value of a after the command:
  - Serial.println(++a);
  - Serial.println(a++);
  - Serial.println(--a);
  - Serial.println(a--);
- What would be the output of:
  - int a;
  - 3=a;
- Length and breadth of a rectangle are 5 and 7 respectively. Write a program to calculate the area and perimeter of the rectangle.

Left to Students (C & Python)

- Length and breadth of a rectangle are 5 and 7 respectively. Write a program to calculate the area and perimeter of the rectangle.
- Write a program to determine whether the number from the serial port is even or odd. If we send 257, it sends back «odd». If we send 248, it sends back «even».
- Write a program to reverse a 3-digit number which is entered from the serial port. If we send 257, it sends back 752.
- Write a program to calculate the sum of the digits of a 3-digit number which is entered from the serial port. If we send 257, it sends back 14.

Thanks for listening ☺

YALÇIN İŞLER

Assoc. Prof.