Unary Operator

C Operation	Operator	Example
Positive	+	a=+3
Negative	-	b=-a
Increment	++	i++
Decrement	--	i--

- The first assigns positive 3 to a
- The second assigns the negative value of a to b.
- i++ is equivalent to i = i + 1
- i-- is equivalent to i = i-1

PRE- / POST-Increment

- It is also possible to use ++i and --i instead of i++ and i--
- However, the two forms have a slightly yet important difference.
- Consider this example:

```
int a = 9;
printf("%d\n", a++);
printf("%d", a);
```

- The output would be:

9
10

PRE- / POST-Increment cont…

- But if we have:

```
int a = 9;
printf("%d\n", ++a);
printf("%d", a);
```

- The output would be:

  10
  10

- a++ would return the current value of *a* and then increment the value of *a*
- ++a on the other hand increment the value of *a* before returning the value

The following table illustrates the difference between the prefix and postfix modes of the increment and decrement operator.

```
int R = 10, count=10;
```

++ Or -- Statement	Equivalent Statements	R value	Count value
R = count++;	R = count; count = count + 1	10	11
R = ++count;	count = count + 1; R = count;	11	11
R = count --;	R = count; count = count – 1;	10	9
R = --count;	Count = count – 1; R = count;	9	9

Binary Operators

C Operation	Operator	Example
Addition	+	a+3
Subtraction	-	a-6
Multiplication	*	a*b
Division	/	a/c
Modulus	%	a%x

- The division of variables of type int will always produce a variable of type int as the result.
- You could **only use** modulus (%) operation on int variables.

Assignment Operators

- Assignment operators are used to combine the '=' operator with one of the binary arithmetic operators
- In the following slide, All operations starting from c = 9

Operator	Example	Equivalent Statement	Results
+=	c += 7	c = c + 7	c = 16
-=	c -= 8	c = c – 8	c = 1
*=	c *= 10	c = c * 10	c = 90
/=	c /= 5	c = c / 5	c = 1
%=	c %= 5	c = c % 5	c = 4

Conditional Operator

- The conditional operator (?:) is used to simplify an if/else statement.
- Syntax:

  Condition ? Expression1 : Expression2
- The statement above is equivalent to:

```
if (Condition)
    Expression1
else
    Expression2
```

Conditional Operator cont…

- Example 1:
  <u>if/else statement:</u>

```
if (total > 60)
        grade = 'P'
else
        grade = 'F';
```

<u>conditional statement:</u>

```
total > 60 ? grade = 'P': grade = 'F';
```

OR

```
grade = total > 60 ? 'P': 'F';
```

Conditional Operator cont…

- Example 2:

  if/else statement:

```
if (total > 60)
        printf("Passed!!\n");
else
        printf("Failed!!\n");
```

  Conditional Statement:

```
printf("%s!!\n", total > 60? "Passed": "Failed");
```

Conclusions on C Operators

- This chapter exposed you the operators used in C
  - Arithmetic operators
  - Assignment operators
  - Equalities and relational operators
  - Logical operators
  - Conditional operator
- Precedence levels come into play when there is a mixed of arithmetic operators in one statement.
- Pre/post fix - effects the result of statement

Selection structure: *switch*

- A *switch* statement is used to choose one choice from multiple cases and one default case.
- Syntax:

```
switch (variable)
{
     case case1:
        statement1;
          break;
     case case2:
        statement2;
        break;
     …
     default;
        statement;
        break;
}
```

The *break* statement is needed so that once a case has been executed, it will skip all the other cases and go outside the *switch* statement.
If the *break* statement is omitted, the execution will be carried out to the next alternatives until the next *break* statement is found.

switch - example

```
int number;

printf("Enter a positive integer number: ");
scanf("%d",&number);

switch (number) {
   case 1:
         printf("One!!\n");
         break;
   case 2:
         printf("Two!!\n");
         break;
   case 3:
         printf("Three!!\n");
         break;
   default:
          printf("Others\n");
          break;
}
```

This program reads a number from the user and print out the string equivalent for 1, 2 or 3.
If the value being keyed in is other than 1,2 or 3, the default statement will be executed where the statement "Others" will be printed out.