

CSE 4107

QUIZ-04

Name: Rahinur Bin Naushad

ID : 220041118

Arithmetic and Logical
expression





Arithmetic operations be like :

$x = x + 1$ in
Computer
Science

$x = x + 1$
in Math



Also logical operation :

$\&\&$!alive	alive
!dead		
dead		

Arithmetic expressions

There are five arithmetic operators in C, which are given below:

Operator	Name of the operator	Arithmetic operation	Syntax
+	Addition	Add two operands.	$x+y$
-	Subtraction	Subtract the second operand from the first operand.	$x-y$
*	Multiplication	Multiply two operands.	$x*y$
/	Division	Divide the first operand by the second operand.	x/y
%	Modulus	Calculate the remainder when the first operand is divided by the second operand.	$x\%y$

N.B

- i) Dividing by zero will crash the program.
- ii) Modulus only works for int (integer) datatype.



**0 in addition and
subtraction**



0 in multiplication



0 in division

Logical expressions

There are 3 three logical operators in C, which are given below:

1. Logical AND operator (&&)

X	Y	X&&Y
0	0	0
1	0	0
0	1	0
1	1	1

2. Logical OR operator (||)

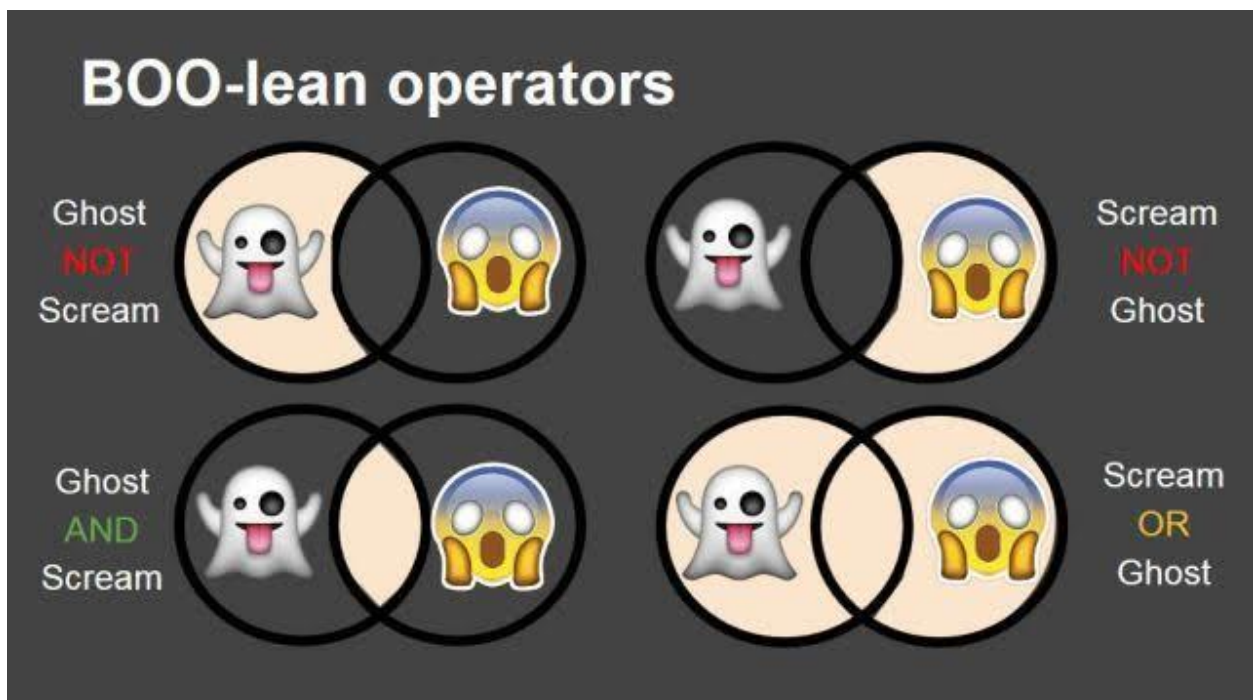
X	Y	X Y
0	0	0
1	0	1
0	1	1
1	1	1

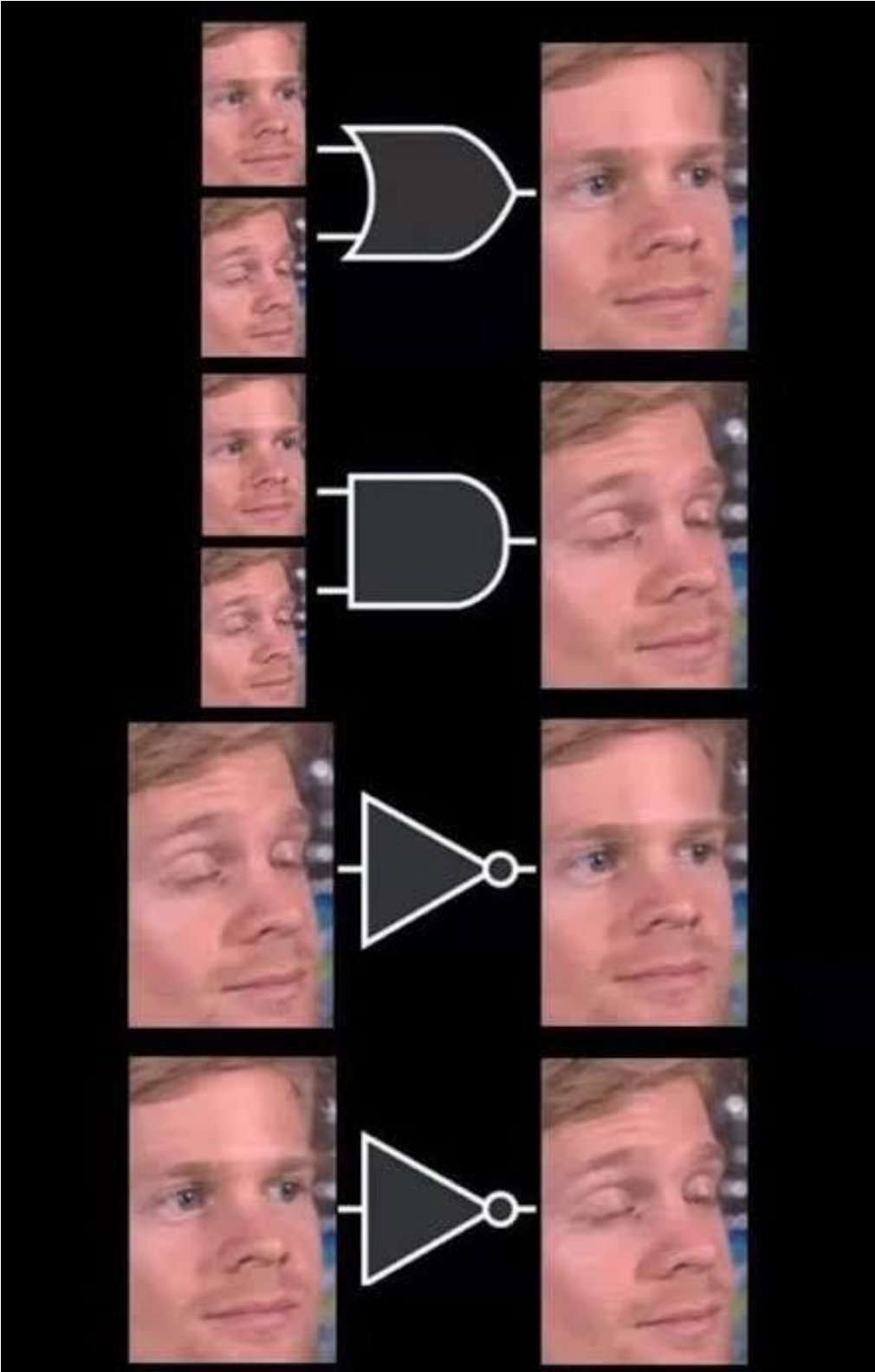
2. Logical NOT operator (!)

X	!X
0	1
1	0

N.B

Here, 0 and 1 represent false and true statements respectively.

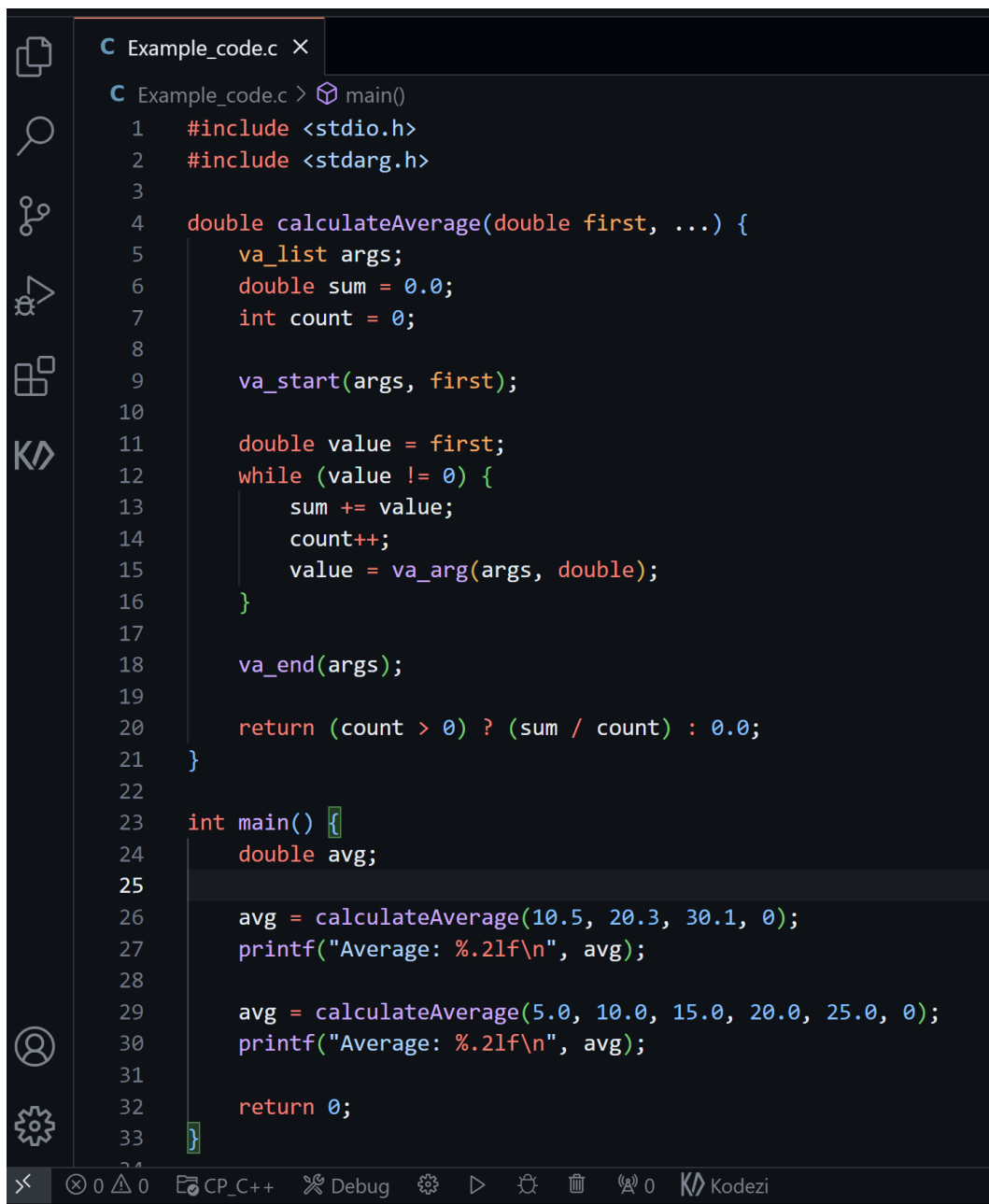




An example problem1

Create a function that takes as many arguments as the user wishes and return their average number.

Solution

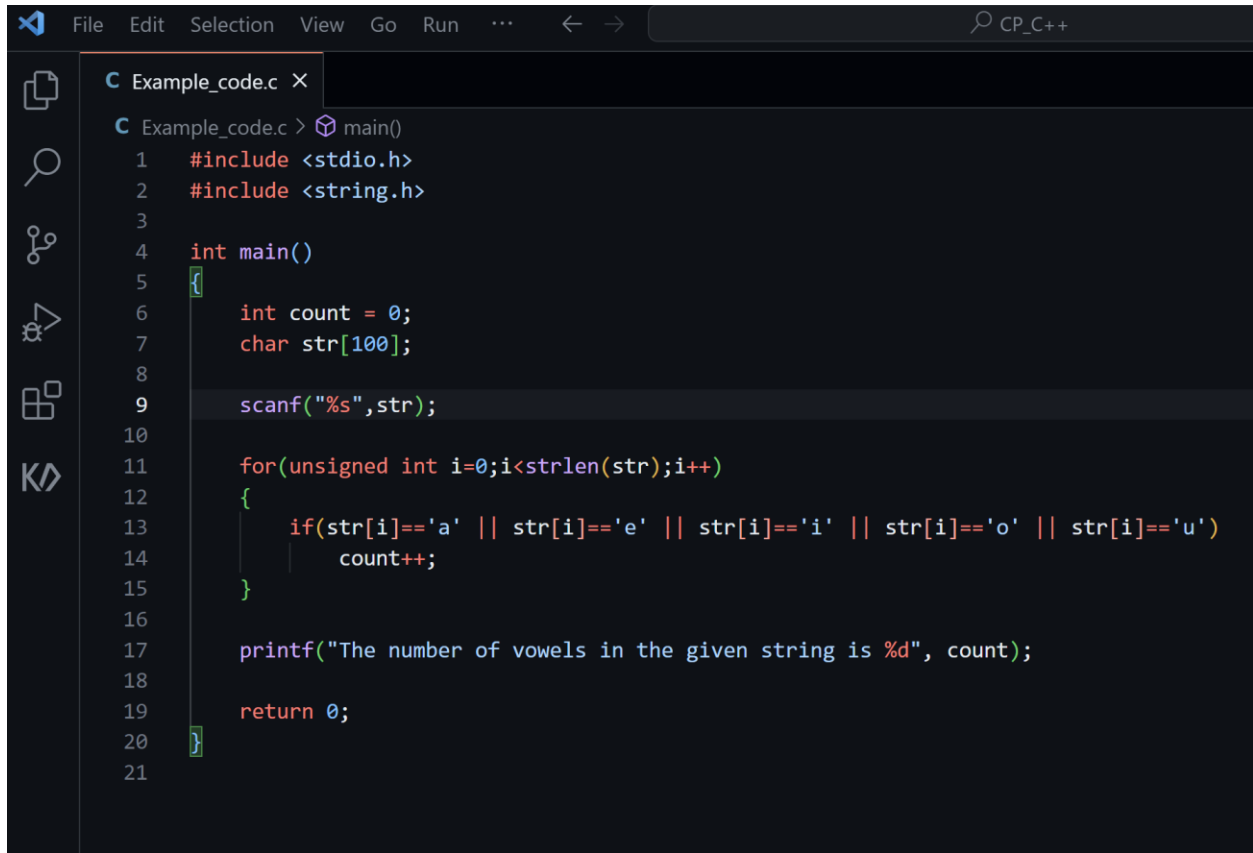


```
C Example_code.c X
C Example_code.c > main()
1  #include <stdio.h>
2  #include <stdarg.h>
3
4  double calculateAverage(double first, ...) {
5      va_list args;
6      double sum = 0.0;
7      int count = 0;
8
9      va_start(args, first);
10
11     double value = first;
12     while (value != 0) {
13         sum += value;
14         count++;
15         value = va_arg(args, double);
16     }
17
18     va_end(args);
19
20     return (count > 0) ? (sum / count) : 0.0;
21 }
22
23 int main() {
24     double avg;
25
26     avg = calculateAverage(10.5, 20.3, 30.1, 0);
27     printf("Average: %.2lf\n", avg);
28
29     avg = calculateAverage(5.0, 10.0, 15.0, 20.0, 25.0, 0);
30     printf("Average: %.2lf\n", avg);
31
32     return 0;
33 }
```


An example problem2

Given a string of lowercase alphabet as an input. Check how many vowels are there in the string.

Solution

A screenshot of a C++ IDE window titled 'CP_C++'. The editor shows a file named 'Example_code.c' with a 'main()' function. The code includes <stdio.h> and <string.h>. It declares an integer 'count' and a character array 'str' of size 100. It uses 'scanf' to read a string into 'str'. A 'for' loop iterates through the string, and an 'if' statement checks for vowels ('a', 'e', 'i', 'o', 'u'). If a vowel is found, 'count' is incremented. Finally, 'printf' prints the total count of vowels, and 'return 0;' ends the program.

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main()
5  {
6      int count = 0;
7      char str[100];
8
9      scanf("%s",str);
10
11     for(unsigned int i=0;i<strlen(str);i++)
12     {
13         if(str[i]=='a' || str[i]=='e' || str[i]=='i' || str[i]=='o' || str[i]=='u')
14             count++;
15     }
16
17     printf("The number of vowels in the given string is %d", count);
18
19     return 0;
20 }
21
```