Programming – 24.09.19

* “getchar()” = workaround for Borland, checks for the user pressing enter. Effective use: keeping the window open.
* *EXAMPLE CODE – Example of incrementing a variable, the ++ can be replaced with a “–“ if you wish to subtract. Note this only works when implementing +/- 1.*

Int var;

var = 1;

var = var+1;

Above = Below

var++

* *EXAMPLE CODE – Same for all of them, so \*= or +=, etc.*

var = 2;

var = var \* 3;

OR

Var =\* 3

* Redundant code = code that does nothing for example “var\*\*” which is just 1 by 1 = 1.
* EXAMPLE CODE – One? Showing an example of var1++ and such.

#include (stdio.h)

int main()

{

int var1, var2;

var1 = 1;

var2 = 2;

printf(“var1 is %d and var2 is %d”, var1, var2);

var1++;

var2--;

printf(“var1 is %d and var2 is %d”, var1, var2);

return 0;

}

* EXAMPLE CODE: - Showing a more confusing incrementation example.

#include (stdio.h)

Int main()

{

Int var1, var2, var3, var4

var1 = var2 = 1;

var3 = var1++:

var4 = ++var2;

printf(“ var1 %d and var2 %d \n”, var1, var2);

printf(“ var3 %d and var4 %d \n”, var3, var4);

return 0;

}

* REMEMBER var1++ is an *INCREMENTATION.* In the code var1 = var2++, var 1 does NOT become 2. Var1 remains 1, while var2 increments by one.
* EXAMPLE CODE:

#include (stdio.h)

Int main()

{

num = 2 + 4 \* 3;

* Remember, computers obey BIMDAS.
* If there are two of the same level, it follows the order of their arrangement. (BIMDAS again).
* If you want to force a computer to not follow BIMDAS, use brackets. (Duh).