

TRABAJO PRÁCTICO N° 1

Deben entregarse por grupos en la entrega correspondiente vía Campus los archivos .c correspondientes a los ejercicios indicados.

1. Prediga la salida del siguiente programa, verifique y justifique los resultados obtenidos.

```
#include <stdio.h>

int main(void)
{
    int a = 10;
    printf("%d", a);
    a+1;
    printf("%d", a);
    return 0;
}
```

2. Prediga la salida del siguiente programa, verifique y justifique los resultados obtenidos.

```
#include <stdio.h>

int main(void)
{
    int x = 5;

    printf("Is x equal to 6: %c\n", x = 6 ? 'Y' : 'N');

    return 0;
}
```

3. Prediga la salida del siguiente programa, verifique y justifique los resultados obtenidos.

```
#include <stdio.h>

int main (void)
{
    double d;
    double half = 1/2;
    int x = 5, y = 2;
    d = x/y;

    printf("%f\n", half);
    printf("%f\n", d);

    return 0;
}
```

¿Cómo lo modificaría para obtener un resultado matemáticamente deseable?

4. Predecir el valor de las siguientes expresiones y luego comprobar las mismas.

```
#include <stdio.h>
int main (void)
{
    int x = 100, y = 50;

    printf ("%d", x);
    printf ("%d", x++);
    printf ("%d", --x);
    printf ("%d", x--);
    printf ("%d", ++x);

    printf ("%X", x);
    printf ("%X", x & 0xF4 == 0);
    printf ("%X", x == 0xF4 & 0xFF);
    printf ("%X", x>2 && 0xF4);
    printf ("%X", (32%9%4));
    printf ("%d", x>3 && x>5);
    printf ("%d", x>3 || x>5 || x++);

    printf ("%d", x);
    printf ("%d", x>3 && x>520 && x++);

    printf ("%d", x);
    printf ("%d", y += x);
    printf ("%d", y *= x);
    return 0;
}
```

5. Explique cuál será la salida del siguiente programa y verifíquelo.

```
#include <stdio.h>
void main(void)
{
    unsigned char x = 255;
    signed char y = -1;

    printf("x= %d\n y = %d\n", x>>2, y>>2);

    return 0;
}
```

6. **[ENTREGAR]** Defina una variable porta suponiendo que representa el puerto A del HC11 y asígnele un valor. Se pide crear un programa que realice lo siguiente:

- Imprima un 1 si los bits 5 y 1 del puerto están en 1 y/o si los bits 2 y 6 del puerto están en 0;
- En caso contrario imprima 0.

En todos los casos definir máscaras usando *#define*.

7. Dadas las mismas condiciones del ejercicio anterior, imprimir los números contrarios (cero en el caso a, uno en el caso b). ¿Hubo que hacer muchos cambios? ¿Hay alguna manera de implementarlo de manera de que se pueda cambiar lo que se imprime rápidamente?

8. **[ENTREGAR]** Escribir un programa que dada una variable tipo *double* calcule (es decir, genere una nueva variable con el valor) e imprima:

- a. el entero superior (función techo)¹,
- b. el entero inferior (función piso)¹,
- c. el número redondeado a un entero,
- d. la parte entera del número (función truncado)¹,
- e. el módulo del número (valor absoluto).

*ACLARACIÓN: La consigna pide que se genere una variable con el resultado, es decir, por cada punto debe haber un **double** con el valor correspondiente. No alcanza con hacer el redondeo a través de los especificadores de formato de **printf()**. Revisar bien las definiciones matemáticas de los valores pedidos en el link de Wikipedia que se encuentra al final de esta página.*

¹ https://es.wikipedia.org/wiki/Funciones_de_parte_entera