

Práctica 1

Semestre de otoño

Curso 2019/20

- **Fase 1:**

- 1.- Crear en el explorador de archivos la carpeta ***I:/progl***.
- 2.- Crear en el explorador de archivos la carpeta ***P1*** en ***I:/progl***.
- 3.- Abrir *Codeblocks* y crear un proyecto vacío de nombre ***P1F1*** en ***I:/progl/P1***
- 4.- Seleccionando **File -> New -> Empty File** y añadiendo al proyecto este fichero con el nombre ***p1f1.c*** se escribirá en la ventana del editor el siguiente programa:

```
#include <stdio.h>

int main(void)
{
    printf("Hello world!\n");
    return 0;
}
```

Una vez que el fichero fuente se ha editado, proceder a compilarlo (seleccionado la opción *Build* en el desplegable de la opción del mismo nombre en la barra de tareas) y ejecutar el programa.

- **Fase 2:**

Se facilita el código fuente, sin errores, de un sencillo programa que realiza operaciones de suma. Se deberán realizar distintas sumas y observar qué ocurre cuando las cifras quedan fuera del rango de las variables utilizadas para definir los operandos. **En el comentario de cabecera del programa se indican las pruebas a realizar y ese mismo espacio es el destinado para escribir las respuestas.**

- 1.- Abrir *Codeblocks* y crear un proyecto vacío ***P1F2*** en ***I:/progl/P1***
- 2.- Seleccionando **File -> New -> Empty File** y añadiendo al proyecto este fichero con el nombre ***p1f2.c*** se podrá copiar en la ventana del editor el código facilitado al alumno en el fichero ***p1f2.c***. (Ctrl+C, Ctrl+V).

Otra manera de añadir el código fuente al proyecto es guardar el fichero facilitado ***p1f2.c*** en

I:/progl/P1/P1F2

y añadir ese fichero, con la opción *Add Files* que aparece en el desplegable al pulsar el botón derecho sobre el nombre del proyecto en la ventana de *management*.

En las siguientes fases se seguirá el mismo proceso en cuanto a creación de proyectos y ficheros fuentes que para las fases 1 y 2.

- **Fase 3:**

Se facilita el código fuente, con errores de compilación, de un sencillo programa que realiza operaciones de suma, resta, multiplicación y división. Se deberán corregir los errores de compilación y comprobar el funcionamiento. **En el comentario de cabecera del programa se escribirán los números de las líneas de código en las que se han encontrado errores de compilación y cómo se han solucionado.**

Prueba qué sucede cuando se divide 25 entre 0. **Modifica el programa** para que su funcionamiento sea adecuado e imprima el siguiente mensaje de error "Error: el divisor es 0" y no realice la operación cuando el divisor sea 0.

TERCERA PARTE. Manejo del depurador


- **Fase 4:**

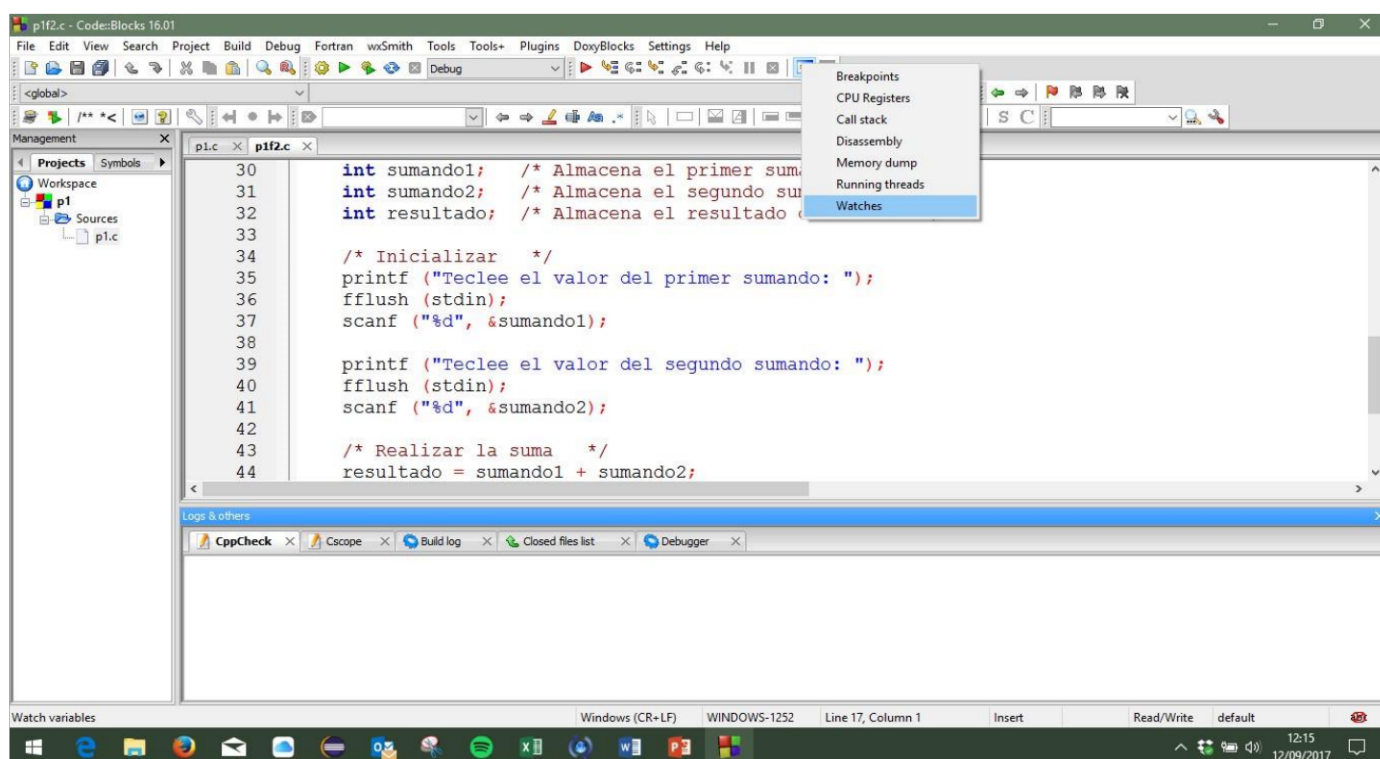
Se facilita el código fuente de un programa que calcula un producto por sumas sucesivas.

El programa se debe ejecutar con el depurador paso a paso analizando la evolución del valor de las variables.

En la cabecera del programa se escribirá la evolución de estos valores.

Para manejar la funcionalidad del depurador de CodeBlocks ayúdese del documento *Tutorial de CodeBlocks* disponible en Moodle. Se debe completar la tabla con los valores de las variables que aparece en el comentario de cabecera del programa.

Pinchando en el icono del **debugger**  y seleccionando **watches** se puede ir haciendo un seguimiento del valor de cada variable del programa.



- **Fase 5: Ampliación tipos de datos.** Modifica el programa de la fase 3 para operar con números reales.
- **Fase 6: Codificación por analogía.** Modificar el programa de la Fase 4 para que calcule la potencia de un número elevado a otro por multiplicaciones sucesivas.

Se creará un *proyecto* diferente para cada práctica. Si la práctica tiene más de una fase, se creará un proyecto para cada fase de la práctica. El nombre de cada proyecto será *Px* siendo x el número de la práctica (si sólo hay una fase en el desarrollo de la práctica), o *PxFy* siendo y el número de la fase de la práctica x.

El nombre del fichero fuente contenido en cada proyecto será *pxfy.c*

Así, el proyecto de la fase 1 de la práctica 1 será “P1F1”, y el fichero fuente será “p1f1.c”.

Nombre: **p1f2.c**
 Autor: Programación I
 Fecha: Curso 2019-2020
 Algoritmo: Realiza la suma de dos sumandos enteros y presenta el resultado por pantalla

ACCIONES A REALIZAR: Crea el proyecto y genera el fichero ejecutable.

Prueba el programa y rellena el valor del resultado, explicando el motivo en los casos en los que solicite

7+25=
 7-25=
 2147483647+3=
 Explicación:

7.25+25.75=
 Explicación:

*****/

```
#include <stdio.h>
```

```
int main ( void )
{
    int sumando1;    /* Almacena el primer sumando */
    int sumando2;    /* Almacena el segundo sumando */
    int resultado;   /* Almacena el resultado de la suma */

    /* Inicializar */
```

```

printf ("Teclee el valor del primer sumando: ");
fflush (stdin);
scanf ("%d", &sumando1);

printf ("Teclee el valor del segundo sumando: ");
fflush (stdin);
scanf ("%d", &sumando2);

/* Realizar la suma */
resultado = sumando1 + sumando2;

/* Presentar resultados */
printf ("\nEl resultado de la suma es %d \n", resultado);

return 0;
}

```

```

/*****

```

```

Nombre:    p1f3.c
Autor:     Programación I
Fecha:     Curso 2019-2020

```

ACCIONES A REALIZAR: Crea el proyecto y compila el programa.
 Corrige los errores sintácticos que te indica el compilador.
 Genera el fichero ejecutable y prueba el programa.
 Añade el código necesario para evitar el error de ejecución
 que tiene lugar en la división por 0.
 Errores encontrados y corregidos:
 Nº de línea.....Error.....Corrección.....

```

*****/

```

```

#include <stdio.h>

```

```

int main ( void )
{
    int operando1;    /* Almacena el primer operando */
    int operando2     /* Almacena el segundo operando */
    int resultado;     /* Almacena el resultado de las operaciones */

    /* Inicializar */
    printf ("Teclee el valor del primer operando: ");
    fflush (stdin);
    scanf ("%d", &operando1);

    printf ("Teclee el valor del segundo operando: ");
    fflush (stdin);
    scanf ("%d", &operando2);

    /* Realizar la suma */
    resultado = operando1 + operando2

```

```

/* Presentar resultados */
printf ("\nEl resultado de la suma es %d ", resultado);

/* Realizar la resta */
resultado = operando1 - operando2;

/* Presentar resultados */
printf ("\nEl resultado de la resta es %d ", &resultado);

/* Realizar la multiplicación */
resultado = operando1 * operando2;

/* Presentar resultados */
printf ("\nEl resultado de la multiplicacion es %d ", resultado);

/* Realizar la división */
resultado = operando1 / operando2;

/* Presentar resultados */
printf ("\nEl resultado de la división es %d ", resultado);

return 0;
}

```

/*****

Nombre: **p1f4.c**
 Autor: Programación I
 Fecha: Curso 2019-2020
 Algoritmo: Realiza la multiplicación usando un algoritmo de sumas sucesivas

ACCIONES A REALIZAR: Crea el espacio de trabajo y genera el fichero ejecutable.

Prueba el programa, ejecutándolo paso a paso, para los valores 5 y 3 y completa la siguiente tabla para cada paso que modifique el valor de una variable:

MULTIPLICANDO	MULTIPLICADOR	CONTADOR	RESULTADO
		0	
			0
5			
	3		

AÑADE LAS LÍNEAS NECESARIAS

*****/

```

#include <stdio.h>

int main (void)
{
    int multiplicando;    /* primer operando de la operación */
    int multiplicador;    /* segundo operando de la operación */
    int resultado;        /* resultado de la multiplicación */
    int contador;        /* contador de número de veces que se hace la operación */

    contador = 0;
    resultado = 0;
    printf("Escriba el multiplicando: ");
    fflush (stdin);
    scanf("%d",&multiplicando);
    printf("Escriba el multiplicador: ");
    fflush (stdin);
    scanf("%d",&multiplicador);

    while (contador < multiplicador)
    {
        resultado = resultado+multiplicando;
        contador = contador+1;
    }
    printf("El resultado de multiplicar %d por %d es %d\n", multiplicando,
multiplicador, resultado);

    return 0;
}

```