

Puntos a cumplir en el proyecto Navaja Valirya 2.0 (3ª Evaluación)

1. Los proyectos serán entregados dentro del plazo establecido. ✓

Control de versiones:

2. Todos los cambios serán subidos a Github, incluidos documentos y soluciones. ✓

Refactorización:

3. Usar los estilos de codificación definidos para cambiar los nombres de clases, controles gráficos y variables, con el fin de aumentar el nivel de comprensión. ✓
4. Quitar los inputbox usados, modificando los formularios para que la información se introduzca mediante cajas de texto. ✓
 - 4.1. Limitar la longitud de las cajas de texto. ✓
 - 4.2. Validar la entrada de datos antes de llamar a los métodos de la lógica de negocio, indicando al usuario dónde está el error y cómo resolverlo. ✓
5. Crear clases con la lógica de negocio de cada subsistema. ✓
 - 3.1.1. Las clases estarán en el mismo namespace del subsistema. ✓
 - 3.1.2. El nombre de la clase incluirá la palabra Lógica. ✓
 - 3.1.3. Los métodos serán static en la mitad de los casos. La otra mitad será necesario instanciar la clase para que se puede acceder a sus métodos. ✓
 - 3.1.4. Usa excepciones para comunicar errores de la lógica al interfaz y try/catch para recuperarlos en los formularios. ✓
 - Parámetros tipo out o salidas de métodos para devolver resultados. ✓
 - Extra +1 punto: Diseña nuevas excepciones personalizadas para facilitar el control de errores o usa alternativas como la vista en clase, en la última parte del tema de test unitarios.

(no ha sido necesario crear excepciones personalizadas ya que no existen situaciones excepcionales en las aplicaciones).

Redacción de casos de prueba II:

6. Documentación de las pruebas con sus correspondientes casos de prueba y pruebas de caja negra (explicación). Debes crear los casos de pruebas sobre la aplicación. ✓ **(A continuación)**
 - 6.1. Debe ser el guion usado para codificar las pruebas. ✓ **(A continuación)**
 - 6.2. Estará incluido dentro de la solución. ✓

Pruebas unitarias:

7. Crear pruebas unitarias en las clases de la lógica de negocio, teniendo en cuenta el documento creado para pruebas. Relaciona los métodos de prueba con los casos de prueba. ✓ **(A continuación)**
 - 7.1. Lanza las pruebas y completa las tablas con los errores encontrados. Debes indicar si has encontrado un error en la tabla de casos de uso. ✓ **(A continuación)**
 - 7.2. Resuélvelos y modifica la tabla de casos de pruebas. ✓
(no ha hecho falta ya que no se han encontrado errores).
8. Lanza las pruebas de nuevo y comprueba que todo está correcto. ✓
(no ha hecho falta ya que no se han encontrado errores).
 - 8.1. Haz una copia de la tabla indicando si te ha funcionado al volver a ejecutarlo. ✓
(no ha hecho falta ya que no se han encontrado errores).

Extras +2 puntos. Refactorización:

9. Probar las funcionalidades de análisis de código de Visual Studio para mejorar el código. ✓
 - 9.1. Ver el video y aplicarlo al proyecto. ✓ **(A continuación)**
 - 9.2. Ejecutar el análisis de código de forma manual. Si no encuentras advertencias, cambia el conjunto de reglas. ✓ **(A continuación)**
 - 9.3. Resolver, al menos, dos de las advertencias diferentes de las que te indica, documentando cómo lo has hecho. ✓ **(A continuación)**

- Contenidos relacionados para ampliación (Visual Studio .net Versiones de prueba):
 - Probar la interfaz de usuario con pruebas de IU automatizadas. ✗
 - Usar cobertura de código para determinar la cantidad de código que se está probando. ✓

NO DISPONGO DEL VISUAL STUDIO ENTERPRISE, PERO HE LEIDO LA DOCUMENTACIÓN APORTADA (en el VS Community no existe esa posibilidad).

- Generar pruebas unitarias para el código con IntelliTest. ✓

NO DISPONGO DEL VISUAL STUDIO ENTERPRISE, PERO HE LEIDO LA DOCUMENTACIÓN APORTADA (en el VS Community no existe esa posibilidad).

6-. Documentación de las pruebas con sus correspondientes casos de prueba y pruebas de caja negra (explicación). Debes crear los casos de pruebas sobre la aplicación.

6.1) Debe ser el guion usado para codificar las pruebas.

- Prueba de Caja Negra de la Aplicación 1 CambioDivisa:

Para las presentes pruebas se ha limitado la entrada de datos del siguiente modo:

- Longitud máxima de la caja de texto 15 caracteres.
- Solo admite tipo de datos *double* mayores que 0.
- No permite ejecutar la lógica si la caja de texto está vacía.

Pruebas para la funcionalidad de Cambia a Euros:

Clases de equivalencia:

- Prueba 1: Una cadena vacía.
- Prueba 2: Introducir cualquier carácter que no sea un dato tipo double.
- Prueba 3: Máximo número posible.
- Prueba 4: Máximo número menos 1.
- Prueba 5: Máximo número más 1.
- Prueba 6: Mínimo número posible.
- Prueba 7: Un espacio y un número.
- Prueba 8: Un número y un espacio.
- Prueba 9: Un número, un espacio y otro número.
- Prueba 10: Un número con decimales.
- Prueba 11: Número Cero.
- Prueba 12: Un número negativo.

Fecha: 14/05/2017 Autor: Tomás Escudero

PRUEBA	ENTRADA	SALIDA ESPERADA	SALIDA OBTENIDA	COMENTARIO
1	""	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
2	"a"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
3	"9999999999999999"	"El cambio son: 6010121043837.8174€"	"El cambio son: 6010121043837.8174€"	La salida es CORRECTA
4	"9999999999999998"	"El cambio son: 6010121043837.8105€"	"El cambio son: 6010121043837.8105€"	La salida es CORRECTA
5	"9999999999999999"	"El cambio son: 6010121043837.8174€"	"El cambio son: 6010121043837.8174€"	La salida es CORRECTA
6	"1"	"El cambio son: 0.0060101210438378233€"	"El cambio son: 0.0060101210438378233€"	La salida es CORRECTA
7	" 1"	"El cambio son: 0.0060101210438378233€"	"El cambio son: 0.0060101210438378233€"	La salida es CORRECTA
8	"1 "	"El cambio son: 0.0060101210438378233€"	"El cambio son: 0.0060101210438378233€"	La salida es CORRECTA
9	"1 1"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
10	"1000,5"	"El cambio son: 6.0131261043597419€"	"El cambio son: 6.0131261043597419€"	La salida es CORRECTA
11	"0"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
12	"-1000"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA

Análisis de los resultados.

Se cumplen correctamente todas las clases de equivalencias. Y no se ha detectado ningún error.

En la prueba 5, no permite introducir el siguiente número ya que la caja de texto está limitada a 9 caracteres, por ello, la prueba la realiza con 9 dígitos tal y como la lleva acabo la prueba 3, por lo tanto, la salida es CORRECTA.

Pruebas para la funcionalidad de Cambio a Pesetas:

Clases de equivalencia:

- Prueba 1: Una cadena vacía.
- Prueba 2: Introducir cualquier carácter que no sea un dato tipo double.
- Prueba 3: Máximo número posible.
- Prueba 4: Máximo número menos 1.
- Prueba 5: Máximo número más 1.
- Prueba 6: Mínimo número posible.
- Prueba 7: Un espacio y un número.
- Prueba 8: Un número y un espacio.
- Prueba 9: Un número, un espacio y otro número.
- Prueba 10: Un número con decimales.
- Prueba 11: Número Cero.
- Prueba 12: Un número negativo.

Fecha: 14/05/2017 Autor: Tomás Escudero

PRUEBA	ENTRADA	SALIDA ESPERADA	SALIDA OBTENIDA	COMENTARIO
1	""	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
2	"a"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
3	"9999999999999999"	"El cambio son: 1.6638599999999984E+ 17ptas."	"El cambio son: 1.6638599999999984E+ 17ptas."	La salida es CORRECTA
4	"9999999999999998"	"El cambio son: 1.6638599999999965E+ 17 ptas."	"El cambio son: 1.6638599999999965E+ 17 ptas."	La salida es CORRECTA
5	"9999999999999999"	"El cambio son: 1.6638599999999984E+ 17ptas."	"El cambio son: 1.6638599999999984E+ 17ptas."	La salida es CORRECTA
6	"1"	"El cambio son: 166.386ptas."	"El cambio son: 166.386ptas."	La salida es CORRECTA
7	" 1"	"El cambio son: 166.386ptas."	"El cambio son: 166.386ptas."	La salida es CORRECTA
8	"1 "	"El cambio son: 166.386ptas."	"El cambio son: 166.386ptas."	La salida es CORRECTA
9	"1 1"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
10	"1000,5"	"El cambio son: 166469.193 ptas."	"El cambio son: 166469.193 ptas."	La salida es CORRECTA
11	"0"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
12	"-1000"	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA

Análisis de los resultados.

Se cumplen correctamente todas las clases de equivalencias. Y no se ha detectado ningún error.

En la prueba 5, no permite introducir el siguiente número ya que la caja de texto está limitada a 9 caracteres, por ello, la prueba la realiza con 9 dígitos tal y como la lleva acabo la prueba 3, por lo tanto, la salida es CORRECTA.

- Prueba de Caja Negra de la Aplicación 2 DisposicionEfectivo:

Para las presentes pruebas se ha limitado la entrada de datos del siguiente modo:

- Longitud máxima de la caja de texto 9 caracteres.
- Solo admite tipo de datos *int* mayores que 0.
- No permite ejecutar la lógica si la caja de texto está vacía.

Pruebas para la funcionalidad de Disposición de Efectivo:

Clases de equivalencia:

- Prueba 1: Una cadena vacía.
- Prueba 2: Introducir cualquier carácter que no sea un dato tipo *int*.
- Prueba 3: Máximo número posible.
- Prueba 4: Máximo número menos 1.
- Prueba 5: Máximo número más 1.
- Prueba 6: Mínimo número posible.
- Prueba 7: Un espacio y un número.
- Prueba 8: Un número y un espacio.
- Prueba 9: Un número, un espacio y otro número.
- Prueba 10: Un número con decimales.
- Prueba 11: Número Cero.
- Prueba 12: Un número negativo.
- Prueba 13: Un número que necesite al menos un elemento de cada (billete y moneda) para la solución.

Fecha: 14/05/2017 Autor: Tomás Escudero

PRUEBA	ENTRADA	SALIDA ESPERADA	SALIDA OBTENIDA	COMENTARIO
1	""	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
2	"a"	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
3	"999999999"	"La disposición es: Entregar 99999 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 2 billete/s de 2.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 4 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 2 moneda/s de 10 pts. Entregar 4 moneda/s de 1 pts."	"La disposición es: Entregar 99999 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 2 billete/s de 2.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 4 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 2 moneda/s de 10 pts. Entregar 4 moneda/s de 1 pts."	La salida es CORRECTA
4	"999999998"	"La disposición es: Entregar 99999 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 2 billete/s de 2.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 4 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 2 moneda/s de 10 pts. Entregar 3 moneda/s de 1 pts."	"La disposición es: Entregar 99999 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 2 billete/s de 2.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 4 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 2 moneda/s de 10 pts. Entregar 3 moneda/s de 1 pts."	La salida es CORRECTA

5	"999999999"	"La disposición es: Entregar 99999 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 2 billete/s de 2.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 4 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 2 moneda/s de 10 pts. Entregar 4 moneda/s de 1 pts."	"La disposición es: Entregar 99999 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 2 billete/s de 2.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 4 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 2 moneda/s de 10 pts. Entregar 4 moneda/s de 1 pts."	La salida es CORRECTA
6	"1"	"La disposición es: Entregar 1 moneda/s de 1 pts."	"La disposición es: Entregar 1 moneda/s de 1 pts."	La salida es CORRECTA
7	" 1"	"La disposición es: Entregar 1 moneda/s de 1 pts."	"La disposición es: Entregar 1 moneda/s de 1 pts."	La salida es CORRECTA
8	"1 "	"La disposición es: Entregar 1 moneda/s de 1 pts."	"La disposición es: Entregar 1 moneda/s de 1 pts."	La salida es CORRECTA
9	"1 1"	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
10	"1000,5"	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
11	"0"	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA
12	"-18691"	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	"No ha introducido un valor correcto, por favor, introduzca un número positivo mayor que 0."	La salida es CORRECTA

13	"18691"	"La disposición es: Entregar 1 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 1 billete/s de 2.000 pts. Entregar 1 billete/s de 1.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 1 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 1 moneda/s de 10 pts. Entregar 1 moneda/s de 5 pts. Entregar 1 moneda/s de 1 pts."	"La disposición es: Entregar 1 billete/s de 10.000 pts. Entregar 1 billete/s de 5.000 pts. Entregar 1 billete/s de 2.000 pts. Entregar 1 billete/s de 1.000 pts. Entregar 1 moneda/s de 500 pts. Entregar 1 moneda/s de 100 pts. Entregar 1 moneda/s de 50 pts. Entregar 1 moneda/s de 25 pts. Entregar 1 moneda/s de 10 pts. Entregar 1 moneda/s de 5 pts. Entregar 1 moneda/s de 1 pts."	La salida es CORRECTA
----	---------	---	---	---------------------------------

Análisis de los resultados.

Se cumplen correctamente todas las clases de equivalencias. Y no se ha detectado ningún error.

En la prueba 5, no permite introducir el siguiente número ya que la caja de texto está limitada a 9 caracteres, por ello, la prueba la realiza con 9 dígitos tal y como la lleva acabo la prueba 3, por lo tanto, la salida es CORRECTA.

- Prueba de Caja Negra de la Aplicación 3 FrasePalindromica:

Para las presentes pruebas se ha limitado la entrada de datos del siguiente modo:

- Longitud máxima de la caja de texto 100 caracteres.
- No permite ejecutar la lógica si la caja de texto está vacía.

Pruebas para la funcionalidad de ¿Es Palindrómica?:

Clases de equivalencia:

- Prueba 1: Una cadena vacía.
- Prueba 2: Un carácter.
- Prueba 3: Un carácter y espacio.
- Prueba 4: Un espacio y un carácter.
- Prueba 5: Un carácter, espacio y el mismo carácter.
- Prueba 6: Un carácter, dos espacios y el mismo carácter.
- Prueba 7: Una palabra palíndroma.
- Prueba 8: Una palabra palíndroma y un espacio.
- Prueba 9: Un espacio y una palabra palíndroma.
- Prueba 10: Una palabra palíndroma, espacio y la misma palabra.
- Prueba 11: Una palabra palíndroma, dos espacios y la misma palabra.
- Prueba 12: Una palabra no palíndroma.
- Prueba 13: Una palabra no palíndroma y un espacio.
- Prueba 14: Un espacio y una palabra no palíndroma.
- Prueba 15: Una palabra, espacio y la misma palabra.
- Prueba 16: Una palabra, dos espacios y la misma palabra.
- Prueba 17: Introducir el máximo número de caracteres permitido (100).
- Prueba 18: Introducir el máximo número de caracteres permitido menos 1 (99).
- Prueba 19: Introducir el máximo número de caracteres permitido más 1 (101).
- Prueba 20: Un carácter especial.

*Al definir un carácter en las pruebas, se incluyen letras, números y símbolos.

Fecha: 14/05/2017 Autor: Tomás Escudero

PRUEBA	ENTRADA	SALIDA ESPERADA	SALIDA OBTENIDA	COMENTARIO
1	""	"La caja de texto está vacía, por favor, introduzca una frase."	"La caja de texto está vacía, por favor, introduzca una frase."	La salida es CORRECTA
2	"a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
3	"a "	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
4	" a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
5	"a a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
6	"a a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
7	"oso"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
8	"oso "	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
9	" oso"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
10	"oso oso"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
11	"oso oso"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
12	"hola"	"La frase NO es PALINDRÓMICA"	"La frase NO es PALINDRÓMICA"	La salida es CORRECTA
13	"hola "	"La frase NO es PALINDRÓMICA"	"La frase NO es PALINDRÓMICA"	La salida es CORRECTA
14	" hola"	"La frase NO es PALINDRÓMICA"	"La frase NO es PALINDRÓMICA"	La salida es CORRECTA
15	"hola hola"	"La frase NO es PALINDRÓMICA"	"La frase NO es PALINDRÓMICA"	La salida es CORRECTA
16	"hola hola"	"La frase NO es PALINDRÓMICA"	"La frase NO es PALINDRÓMICA"	La salida es CORRECTA
17	100 veces "aaaa...a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
18	99 veces "aaaa...a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA
19	101 veces "aaaa...a"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA.
20	"&"	"La frase es PALINDRÓMICA"	"La frase es PALINDRÓMICA"	La salida es CORRECTA

Análisis de los resultados.

Se cumplen correctamente todas las clases de equivalencias. Y no se ha detectado ningún error.

En la prueba 19, no permite introducir el ultimo carácter ya que la caja de texto está limitada a 100 caracteres, por ello, la prueba la realiza con 100 caracteres tal y como la lleva acabo la prueba 17, por lo tanto, la salida es CORRECTA.

- Prueba de Caja Negra de la Aplicación 4 ContadorPalabras:

Para las presentes pruebas se ha limitado la entrada de datos del siguiente modo:

- Longitud máxima de la caja de texto 100 caracteres.
- No permite ejecutar la lógica si la caja de texto está vacía.

Pruebas para la funcionalidad de Contar Palabras:

Clases de equivalencia:

- Prueba 1: Una cadena vacía.
- Prueba 2: Un carácter.
- Prueba 3: Un carácter y espacio.
- Prueba 4: Un espacio y un carácter.
- Prueba 5: Un carácter, espacio y el mismo carácter.
- Prueba 6: Un carácter, dos espacios y el mismo carácter.
- Prueba 7: Una palabra.
- Prueba 8: Una palabra y un espacio.
- Prueba 9: Un espacio y una palabra.
- Prueba 10: Una palabra, espacio y la misma palabra.
- Prueba 11: Una palabra, dos espacios y la misma palabra.
- Prueba 12: Una palabra, espacio y una palabra distinta.
- Prueba 13: Una palabra, dos espacios y una palabra distinta.
- Prueba 14: Introducir el máximo número de caracteres permitido (100).
- Prueba 15: Introducir el máximo número de caracteres permitido menos 1 (99).
- Prueba 16: Introducir el máximo número de caracteres permitido más 1 (101).
- Prueba 17: Un carácter especial.
- Prueba 18: Un carácter, 4 espacios y el mismo carácter.

*Al definir un carácter en las pruebas, se incluyen letras, números y símbolos.

Fecha: 14/05/2017 Autor: Tomás Escudero

PRUEBA	ENTRADA	SALIDA ESPERADA	SALIDA OBTENIDA	COMENTARIO
1	""	"La caja de texto está vacía, por favor, introduzca una frase."	"La caja de texto está vacía, por favor, introduzca una frase."	La salida es CORRECTA
2	"a"	"El número de cada palabra diferente que hay en la frase es: a=1"	"El número de cada palabra diferente que hay en la frase es: a=1"	La salida es CORRECTA
3	"a "	"El número de cada palabra diferente que hay en la frase es: a=1"	"El número de cada palabra diferente que hay en la frase es: a=1"	La salida es CORRECTA
4	" a"	"El número de cada palabra diferente que hay en la frase es: a=1"	"El número de cada palabra diferente que hay en la frase es: a=1"	La salida es CORRECTA
5	"a a"	"El número de cada palabra diferente que hay en la frase es: a=2"	"El número de cada palabra diferente que hay en la frase es: a=2"	La salida es CORRECTA
6	"a a "	"El número de cada palabra diferente que hay en la frase es: a=2"	"El número de cada palabra diferente que hay en la frase es: a=2"	La salida es CORRECTA
7	"oso"	"El número de cada palabra diferente que hay en la frase es: oso=1"	"El número de cada palabra diferente que hay en la frase es: oso=1"	La salida es CORRECTA
8	"oso "	"El número de cada palabra diferente que hay en la frase es: oso=1"	"El número de cada palabra diferente que hay en la frase es: oso=1"	La salida es CORRECTA
9	" oso"	"El número de cada palabra diferente que hay en la frase es: oso=1"	"El número de cada palabra diferente que hay en la frase es: oso=1"	La salida es CORRECTA
10	"oso oso"	"El número de cada palabra diferente que hay en la frase es: oso=2"	"El número de cada palabra diferente que hay en la frase es: oso=2"	La salida es CORRECTA
11	"oso oso "	"El número de cada palabra diferente que hay en la frase es: oso=2"	"El número de cada palabra diferente que hay en la frase es: oso=2"	La salida es CORRECTA

12	"oso hola"	"El número de cada palabra diferente que hay en la frase es: oso=1 hola=1"	"El número de cada palabra diferente que hay en la frase es: oso=1 hola=1"	La salida es CORRECTA
13	"oso hola "	"El número de cada palabra diferente que hay en la frase es: oso=1 hola=1"	"El número de cada palabra diferente que hay en la frase es: oso=1 hola=1"	La salida es CORRECTA
14	100 veces "aaaa...a"	El número de cada palabra diferente que hay en la frase es: aaaa...a=1	El número de cada palabra diferente que hay en la frase es: aaaa...a=1	La salida es CORRECTA
15	99 veces "aaaa...a"	El número de cada palabra diferente que hay en la frase es: aaaa...a=1	El número de cada palabra diferente que hay en la frase es: aaaa...a=1	La salida es CORRECTA
16	101 veces "aaaa...a"	El número de cada palabra diferente que hay en la frase es: aaaa...a=1	El número de cada palabra diferente que hay en la frase es: aaaa...a=1	La salida es CORRECTA
17	"&"	El número de cada palabra diferente que hay en la frase es: &=1	El número de cada palabra diferente que hay en la frase es: &=1	La salida es CORRECTA
18	"a a"	"El número de cada palabra diferente que hay en la frase es: a=2	"El número de cada palabra diferente que hay en la frase es: a=2	La salida es CORRECTA

Análisis de los resultados

Se cumplen correctamente todas las clases de equivalencias. Y no se ha detectado ningún error.

En la prueba 16, no permite introducir el ultimo carácter ya que la caja de texto está limitada a 100 caracteres, por ello, la prueba la realiza con 100 caracteres tal y como la lleva acabo la prueba 14, por lo tanto, la salida es CORRECTA.

La refactorización del código ha sido correcta ya que se han corregido los errores de las anteriores pruebas.

7-. Crear pruebas unitarias en las clases de la lógica de negocio, teniendo en cuenta el documento creado para pruebas. Relaciona los métodos de prueba con los casos de prueba

7.1) Lanza las pruebas y completa las tablas con los errores encontrados. Debes indicar si has encontrado un error en la tabla de casos de uso.

- Pruebas Unitarias de la Aplicación 1 CambioDivisa:

Tal y como se ha comprobado mediante las pruebas de caja negra:

La lógica no admite una cadena vacía, ni números negativos, ni datos que no sean del tipo *double*, desestima los espacios antes y después del número introducido y los espacios entre números no los considera datos tipo *double*.

Por ello se realizan las siguientes pruebas unitarias:

PRUEBA UNITARIA	DATOS DE ENTRADA	RESULTADO ESPERADO	RESULTADO DE LA PRUEBA	COMENTARIO
<code>public void</code> TestCambiarEuros()	cantidadPesetas = 10000;	resultadoEsperado = 60.101210438378232;	CORRECTA	Con este caso se prueban todos los valores permitidos.
<code>public void</code> TestCambiarPesetas()	cantidadEuros = 6;	resultadoEsperado = 998.316;	CORRECTA	Con este caso se prueban todos los valores permitidos.

Análisis de los resultados

Se cumplen correctamente todas las pruebas. Y no se ha detectado ningún error.

- Pruebas Unitarias de la Aplicación 2 DisposicionEfectivo:

Tal y como se ha comprobado mediante las pruebas de caja negra:

La lógica no admite una cadena vacía, ni números negativos, ni datos que no sean del tipo *int*, desestima los espacios antes y después del número introducido y los espacios entre números no los considera datos tipo *int*.

Por ello se realizan la siguiente prueba unitarias para probar que devuelve al menos un valor de cada billete o moneda:

PRUEBA UNITARIA	DATOS DE ENTRADA	RESULTADO ESPERADO	RESULTADO DE LA PRUEBA	COMENTARIO
<code>public void</code> TestDisposicionEfectivo()	cantidadPesetas = 18691;	"La disposición es: \n" + "Entregar 1 billete/s de 10.000 pts.\n" + "Entregar 1 billete/s de 5.000 pts.\n" + "Entregar 1 billete/s de 2.000 pts.\n" + "Entregar 1 billete/s de 1.000 pts.\n" + "Entregar 1 moneda/s de 500 pts.\n" + "Entregar 1 moneda/s de 100 pts.\n" + "Entregar 1 moneda/s de 50 pts.\n" + "Entregar 1 moneda/s de 25 pts.\n" + "Entregar 1 moneda/s de 10 pts.\n" + "Entregar 1 moneda/s de 5 pts.\n" + "Entregar 1 moneda/s de 1 pts.";	CORRECTA	Con este caso se prueba que devuelve al menos un valor de cada billete o moneda con una entrada que prueba todos los valores permitidos.

Análisis de los resultados

Se cumple correctamente la prueba. Y no se ha detectado ningún error.

- Pruebas Unitarias de la Aplicación 3 FrasePalindromica:

Tal y como se ha comprobado mediante las pruebas de caja negra:

La lógica no admite una cadena vacía y desestima los espacios en cualquier lugar de la frase (incluso entre las letras de una misma palabra).

Por ello se realizan las siguientes pruebas unitarias:

PRUEBA UNITARIA	DATOS DE ENTRADA	RESULTADO ESPERADO	RESULTADO DE LA PRUEBA	COMENTARIO
<code>public void</code> TestFrasePalindromicaCorrecta()	frase = "dabale arroz a la zorra el abad";	resultadoEsperado = true;	CORRECTA	Con este caso se prueban todos los valores permitidos en el caso que sea una frase palindrómica.
<code>public void</code> TestFrasePalindromicaIncorrecta()	frase = "canta la rana";	resultadoEsperado = false;	CORRECTA	Con este caso se prueban todos los valores permitidos en el caso que NO sea una frase palindrómica.

Análisis de los resultados

Se cumplen correctamente todas las pruebas. Y no se ha detectado ningún error.

- Pruebas Unitarias de la Aplicación 4 ContadorPalabras:

Tal y como se ha comprobado mediante las pruebas de caja negra:

La lógica no admite una cadena vacía y desestima los espacios en cualquier lugar de la frase.

Por ello se realizan la siguiente prueba unitaria:

PRUEBA UNITARIA	DATOS DE ENTRADA	RESULTADO ESPERADO	RESULTADO DE LA PRUEBA	COMENTARIO
<code>public void</code> TestContadorPalabras()	frase = "perro perro perro gato gato oso";	resultadoEsperado = "El número de cada palabra diferente que hay en la frase es: \n" + "gato = 2\n" + "oso = 1\n" + "perro = 3\n";	CORRECTA	Con este caso se prueban todos los valores permitidos.

Análisis de los resultados

Se cumple correctamente la prueba. Y no se ha detectado ningún error.

9-. Prueba las funcionalidades de análisis de código de Visual Studio para mejorar el código y resuelve, al menos, dos de las advertencias diferentes de las que te indica, documentando cómo lo has hecho.

9.1, 9.2 y 9.3) Una vez ejecutado el análisis con las **reglas mínimas administradas por Microsoft** me da los siguientes errores (9 líneas):

Código	Descripción	Proyecto	Archivo	Línea	Estado
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	formCambioDivisa.cs	11	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	formDisposicionEfectivo.cs	11	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	formFrasePalindromica.cs	11	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	FrasePalindromicaLogica.cs	7	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	ContadorPalabrasLogica.cs	8	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	formContadorPalabras.cs	12	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	CambioDivisaLogica.cs	7	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	DisposicionEfectivoLogica.cs	7	Activa
CS1587	El comentario XML no está situado en un elemento válido del idioma	NavajaValirya	formPrincipal.cs	11	Activa

La advertencia CS1587 significa que hay un comentario que no está situado correctamente.

Corresponde a los comentarios de los namespace del proyecto que no se deben comentar tal y como indica el siguiente enlace.

<https://docs.microsoft.com/es-es/dotnet/articles/csharp/misc/cs1587>

Al eliminarlos se corrige el error y como están comentados desde el SandCastle mediante el Namespace Summaries, no es necesario comentarlos en el código.

Solucionados todos los errores aparecidos con las reglas mínimas administradas por Microsoft.

Una vez ejecutado el análisis con las **reglas de corrección básicas de Microsoft** me da el siguiente error en diferentes partes del código (36 líneas):

Códi...	Descripción	Proyecto	Archivo	Línea	Estado de supr...
CA1303	El método 'formCambioDivisa.BCambioEnEuros_Click(object, EventArgs)' pasa una cadena literal como parámetro 'text' de una llamada a 'MessageBox.Show(string)'. En su lugar, recupere las siguientes cadenas de una tabla de recursos: "El cambio son: , €.".	NavajaValirya	formCambioDivisa.cs	45	Activa

La advertencia CA1303 significa que se deben usar cadenas de texto para facilitar la traducción.

He corregido los del código de los botones del formulario FormCambioDivisa y los ".Text" del FormCambioDivisa.designer del siguiente modo:

```
private void BCambioEnPesetas_Click(object sender, EventArgs e)
{
    double cantidadEuros, resultadoPesetas;

    cantidadEuros = 0;
    resultadoPesetas = 0;
    try
    {
        if (double.TryParse(TCantidad.Text, out cantidadEuros))
        {
            if (cantidadEuros > 0)
            {
                resultadoPesetas = CambioDivisaLogica.cambiarPesetas(cantidadEuros);
                MessageBox.Show(CadenasTexto.ResultadoCambio + resultadoPesetas + CadenasTexto.Pesetas);
            }
            else
            {
                MessageBox.Show(CadenasTexto.ValorIncorrectoCambioDivisa);
            }
        }
        else
        {
            MessageBox.Show(CadenasTexto.ValorIncorrectoCambioDivisa);
        }
    }
    catch (Exception ExCambioEnPesetas)
    {
        MessageBox.Show(CadenasTexto.ErrorException + ExCambioEnPesetas.Message);
    }
}
```

```

private void BCambioEnEuros_Click(object sender, EventArgs e)
{
    double cantidadPesetas, resultadoEuros;

    cantidadPesetas = 0;
    resultadoEuros = 0;
    try
    {
        if (double.TryParse(TCantidad.Text, out cantidadPesetas))
        {
            if (cantidadPesetas > 0)
            {
                resultadoEuros = CambioDivisaLogica.cambiarEuros(cantidadPesetas);
                MessageBox.Show(CadenasTexto.ResultadoCambio + resultadoEuros + CadenasTexto.Euros);
            }
            else
            {
                MessageBox.Show(CadenasTexto.ValorIncorrectoCambioDivisa);
            }
        }
        else
        {
            MessageBox.Show(CadenasTexto.ValorIncorrectoCambioDivisa);
        }
    }
    catch (Exception ExCambioEnEuros)
    {
        MessageBox.Show(CadenasTexto.ErrorException + ExCambioEnEuros.Message);
    }
}

```

Textos del FormCambioDivisa.designer:

```

// LCantidad
//
this.LCantidad.AutoSize = true;
this.LCantidad.Font = new System.Drawing.Font("Microsoft Sans Serif", 11.25F, System.Drawing.FontStyle.Regular);
this.LCantidad.Location = new System.Drawing.Point(27, 37);
this.LCantidad.Name = "LCantidad";
this.LCantidad.Size = new System.Drawing.Size(70, 18);
this.LCantidad.TabIndex = 1;
this.LCantidad.Text = CadenasTexto.LCantidad;
//
// BCambioEnEuros
//
this.BCambioEnEuros.Font = new System.Drawing.Font("Microsoft Sans Serif", 11.25F, System.Drawing.FontStyle.Regular);
this.BCambioEnEuros.Location = new System.Drawing.Point(217, 158);
this.BCambioEnEuros.Name = "BCambioEnEuros";
this.BCambioEnEuros.Size = new System.Drawing.Size(144, 61);
this.BCambioEnEuros.TabIndex = 2;
this.BCambioEnEuros.Text = CadenasTexto.BCambiarEuros;
this.BCambioEnEuros.UseVisualStyleBackColor = true;
this.BCambioEnEuros.Click += new System.EventHandler(this.BCambioEnEuros_Click);
//
// BCambioEnPesetas
//
this.BCambioEnPesetas.Font = new System.Drawing.Font("Microsoft Sans Serif", 11.25F, System.Drawing.FontStyle.Regular);
this.BCambioEnPesetas.Location = new System.Drawing.Point(12, 158);
this.BCambioEnPesetas.Name = "BCambioEnPesetas";
this.BCambioEnPesetas.Size = new System.Drawing.Size(139, 61);
this.BCambioEnPesetas.TabIndex = 3;
this.BCambioEnPesetas.Text = CadenasTexto.BCambiarPtas;
this.BCambioEnPesetas.UseVisualStyleBackColor = true;
this.BCambioEnPesetas.Click += new System.EventHandler(this.BCambioEnPesetas_Click);
//
// formCambioDivisa
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.BackgroundImage = ((System.Drawing.Image)(resources.GetObject("$this.BackgroundImage")));
this.ClientSize = new System.Drawing.Size(384, 266);
this.Controls.Add(this.BCambioEnPesetas);
this.Controls.Add(this.BCambioEnEuros);
this.Controls.Add(this.LCantidad);
this.Controls.Add(this.TCantidad);
this.Name = "formCambioDivisa";
this.Text = CadenasTexto.CambioDivisa;
this.ResumeLayout(false);
this.PerformLayout();

```

Aplicando las Cadenas de Texto expuestas a continuación:

Nombre	Valor	Comentario
BCambiarEuros	Cambia a €	BCambioEnEuros.Text
BCambiarPtas	Cambia a Ptas.	BCambioEnPesetas
CambioDivisa	Cambio de Divisa	FormCambioDivisa.Text
Euros	€.	Simbolo € (segunda parte del mensaje de salida)
LCantidad	Cantidad:	LCantidad.Text
Pesetas	ptas.	Abreviación de Pesetas (segunda parte del mensaje de salida)
ResultadoCambio	El cambio son:	Primera parte del mensaje de salida
ValorIncorrectoCambioDivisa	No ha introducido valor correcto, por favor, introduzca un número positivo mayor que 0.	Mensaje de introducción incorrecta de datos
ErrorException	Se ha producido un error:	Error de Excepción

Solucionados los errores aparecidos con las reglas mínimas administradas por Microsoft en el formulario *FormCambioDivisa* y *formCambioDivisa.designer*.