Trabajo Práctico Número 1

“Prueba de Software”

Programación Avanzada

## Integrantes:

-Raimondo, Pablo Andrés

-Sapaya, Nicolás Martín

-Queirolo, Fernando

-Laurenti, Ezequiel

## Cursada:

1er Cuatrimestre 2017 – Turno Noche

# Introducción

Mediante este trabajo de Prueba de Software queremos comprobar el funcionamiento de 9 programas diferentes que deberían resolver todos los casos posibles de un mismo problema, “Número Escondido”.

Para ello realizamos un lote de pruebas para observar los posibles errores de cada uno de ellos y además describiremos una posible solución llegado el caso de que el programa no realice una salida correcta frente a una entrada determinada.

**#01 Caso: Búsqueda Literal**

*Descripción:* Se valida si el programa es capaz de encontrar una coincidencia que se encuentra escrita literalmente en el primer número en el mismo orden.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 8  16345678  4  3456 | SI 2  2 3 | SI 2  2 3 | SI 2  2 3 | SI 2  2 3 | SI 2  2 3 | SI  2 2 3 | SI 2  2 3 | NO | SI 2  2 3 | SI 2  3 2 |

**#02 Caso: No Hay Coincidencias**

*Descripción:* Se verifica que el programa determine correctamente que no existen coincidencias.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 8  87654321  4  5462 | NO | NO | NO | NO | NO | NO | NO | NO | NO | NO |

**#03 Caso: Búsqueda desordenada**

*Descripción:* Se valida que el programa sea capaz de encontrar la coincidencia en el primer número, pero los dígitos se encuentran desordenados.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 8  87654123  4  1234 | SI 1  5 | SI 1  5 | SI 1  5 | SI 1  5 | NO | SI  1 5 | SI 1  5 | SI 1  5 | SI 1  5 | SI 1  5 |

**#04 Caso: Un Sólo Digito En Ambos Números**

*Descripción:* Se verifica que el programa encuentre correctamente una coincidencia entre dos números de una cifra**.**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 1  1  1  1 | SI 1  1 | SI 1  1 | SI 1  1 | SI 1  1 | NO | SI  1 1 | SI 1  1 | SI 1  1 | SI 1  1 | SI 1  1 |

**#05 Caso: Múltiples Apariciones**

*Descripción:* Se desea verificar que el programa tome las N coincidencias, todos los dígitos del primer número.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  2222222222  1  2 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 9  1 2 3 4 5 6 7 8 9 | SI  10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 | SI 10  1 2 3 4 5 6 7 8 9 10 |

**#06 Caso: Apariciones superpuestas**

*Descripción:* Se verifica que siempre se busquen coincidencias a partir del próximo número, sin saltearse los que ya coincidieron previamente

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  2222222222  2  22 | SI 9  1 2 3 4 5 6 7 8 9 | SI 9  1 2 3 4 5 6 7 8 9 | SI 9  1 2 3 4 5 6 7 8 9 | SI 9  1 2 3 4 5 6 7 8 9 | SI 8  1 2 3 4 5 6 7 8 | SI  9 1 2 3 4 5 6 7 8 9 | SI 9  1 2 3 4 5 6 7 8 9 | SI 5  1 3 5 7 9 | SI 9  1 2 3 4 5 6 7 8 9 | SI 9  1 2 3 4 5 6 7 8 9 |

**#07 Caso: Más de 9 dígitos en ambos números**

*Descripción:* Se valida que el programa pueda trabajar con más de 9 dígitos en cada número.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 20  1234567898  7654321898  10  8135927468 | SI 5  1 8 9 10 11 | SI 5  1 8 9 10 11 | NO | NO | SI 4  1 8 9 10 | SI  5 1 8 9 10 11 | SI 5  1 8 9 10 11 | SI 2  1 11 | SI 5  1 8 9 10 11 | ROMPE |

**#08 Caso: Comparación hasta N dígitos**

*Descripción:* Se verifica que el programa recorra todo el primer número para encontrar las posibles coincidencias.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  1111111122  2  22 | SI 1  9 | SI 1  9 | SI 1  9 | SI 1  9 | NO | SI  1 9 | SI 1  9 | SI 1  9 | SI 1  9 | SI 1  9 |

**#09 Caso: Fatiga**

*Descripción:* Se verifica que el programa pueda realizar una correcta salida ante un caso extremo de entrada.

*Ver carpeta #09-Fatiga*

**#10 Caso: Búsqueda al revés**

*Descripción:* Se verifica si el programa es capaz de encontrar la coincidencia en sentido inverso, de derecha a izquierda.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  0987227890  4  7890 | SI 2  1 7 | SI 2  1 7 | SI 1  7 | SI 2  1 7 | SI 1  1 | SI  2 1 7 | SI 2  1 7 | SI 1  1 | SI 2  1 7 | SI 2  7 1 |

**#11 Caso: Búsqueda desordenada**

*Descripción:* Se verifica que el programa encuentre correctamente coincidencias con dígitos desordenados.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  0897120897  4  7890 | SI 2  1 7 | SI 2  1 7 | NO | NO | SI 1  1 | SI  2 1 7 | SI 2  1 7 | SI 1  1 | SI 2  1 7 | SI 2  1 7 |

**#12 Caso: Lectura de M-1 consecutivos**

*Descripción:*Se valida si el programa encuentra coincidencias con M-1 dígitos consecutivos

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 12  710982217890  5  78901 | SI 2  1 8 | SI 2  1 8 | SI 1  8 | SI 2  1 8 | SI 1  1 | SI  2 1 8 | SI 2  1 8 | SI 1  1 | SI 2  1 8 | SI 2  1 8 |

**#13 Caso: Caso trivial**

*Descripción:* Caso general para observar el funcionamiento del programa.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  1234560789  7  0123456 | SI 1  1 | SI 1  1 | SI 1  1 | SI 1  1 | SI 1  1 | SI  1 1 | SI 1  1 | SI 1  1 | SI 1  1 | SI 1  1 |

**#14 Caso: Limite de M dígitos**

*Descripción:* Se valida que no existan limitaciones en la cantidad de cifras ingresadas en M, en este caso M >= 8.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 10  1234560789  8  01234567 | SI 1  1 | SI 1  1 | NO | NO | SI 1  1 | SI  1 1 | SI 1  1 | SI 1  1 | SI 1  1 | ROMPE |

**#15 Caso: Múltiples repeticiones desordenadas**

*Descripción:* Se verifica que el programa encuentre las coincidencias incluyendo todos los dígitos aunque ya tuvieran una coincidencia previa o desordenada.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 8  21321231  3  123 | SI 5  1 2 3 5 6 | SI 5  1 2 3 5 6 | SI 2  5 6 | SI 5  1 2 3 5 6 | SI 4  1 2 3 5 | SI  5 1 2 3 5 6 | SI 5  1 2 3 5 6 | SI 1  1 | SI 5  1 2 3 5 6 | SI 5  5 2 1 6 3 |

**#16 Caso: Suma de los M dígitos**

*Descripción:*Se verifica que el programa no tome como valido a una coincidencia tan solo por ser iguales en la suma de sus dígitos. Pueden ser la misma cantidad pero distintos valores y al realizar la adición entre los mismos sea la misma.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| IN | OUT Esperado | A | B | C | D | E | F | G | H | I |
| 7  7205432  3  324 | SI 1  5 | SI 1  5 | SI 1  5 | NO | NO | SI  1 5 | SI 3  1 3 5 | NO | SI 1  5 | SI 1  5 |

### Informe de Fallas:

Resumen:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | #01 | #02 | #03 | #04 | #05 | #06 | #07 | #08 | #09 | #10 | #11 | #12 | #13 | #14 | #15 | #16 |
| A.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| C.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| D.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| E.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| F.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| G.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| H.EXE |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| I.EXE |  |  |  |  |  |  | R |  | R |  |  |  |  | R |  |  |

*Referencias:*

Verde 🡪Salida Correcta Rojo 🡪Salida Incorrecta R🡪 Deja de funcionar el programa

Análisis de fallas cada programa:

* **A.exe:**

No se ha podido detectar fallas.

* **B.exe:**

-Su modo de lectura es sólo de izquierda a derecha (no del sentido contrario).

-El número a buscar (el segundo de M dígitos), como máximo puede dividirse en 2 partes a buscar, estas deben ser literalmente como están escritas, no puede ser forma desordenada.

Ej:

Si el número a buscar es 12345:

Pueden ser los siguientes casos:

\* Que encuentre los 5 dígitos literalmente en ese orden o de las siguientes maneras:

123 45

1 2345

1234 5

12 345

Estos errores pueden ser vistos en **Caso #12**, **Caso #13**, **Caso #14**.

* **C.exe:**

-En el momento en que se busca el segundo conjunto de dígitos en el primero, debe haber M-1 dígitos consecutivos tanto de izquierda a derecha o viceversa, para que sea una salida valida.

Ej:

Siendo el conjunto de dígitos a buscar: 78901, puede encontrarse de las siguientes maneras: 78901, 17890, 10987, 71098.

Estos errores pueden ser vistos en **Caso #07**, **Caso #09**, **Caso #12**.

* **D.exe:**

-La comparación la realiza hasta N-1 dígitos del primer conjunto.

Estos errores pueden ser vistos en **Caso #05**, **Caso #06**, **Caso #08**.

* **E.exe:**

-La cantidad de apariciones que debe figurar en la primera línea de la salida, figura en la segunda.

*Observación del E.exe:*

En varios casos probados la salida del programa era correcta haciendo a un lado el error mencionado anteriormente.

* **F.exe:**

-Suma los valores de los dígitos del segundo número y los compara con la suma de los dígitos del primer número agrupados de a M dígitos.

Ej:

324 = 9, 720 = 9, 054 = 9

Estos errores pueden ser vistos en **Caso #16**.

* **G.exe:**

-Subdivide al conjunto N, en M conjuntos independientes entre sí, no obstante lee cualquier combinación dentro de ese conjunto.

Estos errores pueden ser vistos en **Caso #12**, **Caso #13**, **Caso #14**.

* **H.exe:**

-No se ha podido detectar fallas.

* **I.exe:**

-Si M > = 8, el programa rompe.

Estos errores pueden ser vistos en **Caso #13** y **Caso #14**.

*Observación del I.exe:*

En la salida algunas veces las apariciones no están ordenadas de menor a mayor, esto es debido a que busca exactamente como está el segundo número y luego lo busca desordenado buscando del primer digito al segundo y así sucesivamente, siempre de izquierda a derecha.

Esta observación puede ser vista en **Caso #01**, **Caso #15**.