|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Claudia Rodríguez Espino |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 4 |
| *No de Práctica(s):* | 3 |
| *Integrante(s):* | Navarrete Rodríguez Edwin Daniel |
|  |  |
| *No. de Equipo de cómputo empleado* | 25 |
| *Semestre:* | 2019-2 |
| *Fecha de entrega:* | 27-02-2019 |
| *Obervaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**Práctica 3**

**Guía Práctica de Estudio 03**: Solución de problemas y Algoritmos.

**Objetivo:**

Elaborar algoritmos correctos y eficientes en la solución de problemas siguiendo las etapas de Análisis y Diseño pertenecientes al Ciclo de vida del software.

**Actividades:**

* A partir del enunciado de un problema, identificar el conjunto de entrada y el conjunto de salida.
* Elaborar un algoritmo que resuelva un problema determinado (dado por el profesor), identificando los módulos de entrada, de procesamiento y de salida.

Las actividades propuestas son las siguientes:

* Suma de dos números

**Problema:** Realizar la suma de dos números.

**Datos de entrada:** Dos números números reales.

**Datos de salida:** La suma de los dos números.

**Dominio:** Todos los números reales.

**Solución:**

1. Solicitar dos números.
2. Realizar la suma de los dos números.
3. Mostrar el resultado de la suma.

**Prueba de escritorio**

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | x | y | Salida |
| 1 | 6 | 7 | 13 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | x | y | Salida |
| 1 | -6 | -7 | -13 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | x | y | Salida |
| 1 | -6 | 7 | 1 |

* Área de un círculo

**Problema**: Calcular el área de un círculo.

**Restricciones**: El valor del radio debe ser positivo.

**Datos de entrada**: Radio del círculo (número real).

**Datos de salida**: El área del círculo.

**Dominio**: Todos los números reales positivos.

**Solución:**

1. Solicitar el radio del círculo.
2. Si el radio no es mayor o igual a cero regresar al punto 1.
3. Si el radio es mayor o igual a 0, crear una variable pi = 3.1416
4. Realizar la operación de multiplicación pi por radio al cuadrado: A=pi\*(r\*r)
5. Si el área es igual a cero imprimir “Es un punto”.
6. Si el área es diferente de 0, imprimir el resultado del área del círculo.

**Prueba de escritorio**

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | r | pi | Salida |
| 1 | -1 | 3.1416 | - |
| 2 | 1 | 3.1416 | 3.1416 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | r | pi | Salida |
| 1 | 2 | 3.1416 | 12.5664 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | r | pi | Salida |
| 1 | 0 | 3.1416 | Es un punto |

* Factorial de un número entre 0 y 5

**Problema:** Calcular el Factorial de un número.

**Restricciones**: El número de entrada debe ser igual o mayor a 0 y menor o igual que 5.

**Datos de entrada**: Números enteros positivos del 0 al 5.

**Datos de salida**: La impresión del factorial del número.

**Dominio**: Todos los números enteros positivos.

**Solución:**

1. Solicitar un número entero mayor o igual a 0, y menor o igual a 5.
2. Si el número es menor que cero o mayor a 5, regresar al punto 1.
3. Si el número se encuentra en el intervalo indicado realizar el factorial.

3.1 Si el valor del número es igual a 0, imprimir factorial = 1.

3.2 Si el valor es diferente de 0 introducir una variable contador que comience en 1 y una variable factorial que comience en 1.

3.2.1 Si la variable contador es igual o menor que el número entero de entrada realizar lo siguiente:

3.2.1.1 Multiplicar el valor de la variable contador con el valor de la variable factorial. El resultado se almacena en la variable factorial.

3.2.1.2 Incrementar el valor del contador en uno.

4. Regresar al punto 3.2.1

5. Si el valor de la variable contador no es menor o igual que el valor del número de entrada, imprimir el resultado almacenado en la variable factorial.

**Prueba de escritorio**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteración** | x | factorial | contador | Salida |
| 1 | 0 | 1 | 1 | factorial = 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteración** | x | factorial | contador | Salida |
| 1 | -2 | 1 | 1 | - |
| 2 | -1 | 1 | 1 | - |
| 3 | 2 | 1 | 1 | factorial = 2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteración** | x | factorial | contador | Salida |
| 1 | 3 | 1 | 1 | 6 |

* Determinar si un número es par o impar

**Problema:** Determinar si un número es par o impar.

**Restricciones:** El número de entrada debe ser positivo.

**Datos de entrada:** Número entero.

**Datos de salida:** La evaluación del número ya sea par o impar.

**Dominio**: Todos los números reales positivos.

**Solución:**

1. Solicitar un número entero.
2. Si el número es menor o igual a cero regresar al punto 1.
3. Si el número no es menor o igual a cero realizar la división del número entre 2.
4. Si el residuo es igual a 0, imprimir “número par”.
5. Si el residuo es diferente de 0, imprimir “número impar”.

**Prueba de escritorio**

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | x | Divisor | Salida |
| 1 | 3 | 2 | número impar |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | x | Divisor | Salida |
| 1 | 4 | 2 | número par |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | x | Divisor | Salida |
| 1 | 0 | 2 | - |
| 2 | -1 | 2 | - |
| 3 | 2 | 2 | número par |

**Conclusión:**

En el desarrollo de la práctica se ha logrado comprender el ciclo de vida de software el cual nos permite resolver problemas informáticos mediante una serie de pasos estructurados de manera clara, es decir, el desarrollo de un algoritmo.

Se han propuesto problemas a los cuales se les ha dado solución siguiendo la metodología adecuada de identificación de las necesidades que requiere la tarea dada, el análisis del proceso para averiguar a qué se quiere llegar y definir los conjuntos de entrada y salida del sistema, estableciendo restricciones si lo es requerido.

Los objetivos de esta práctica precisan el proceso de solución de problemas considerando “Definición de necesidades”, “Análisis” y “Diseño”. El proceso de diseño es la generación del algoritmo en el cual se ha propuesto la serie de pasos más concreta y simplificada posible para cumplir con los requerimientos de los problemas propuestos. Se han establecido reglas, restricciones, secuencias de pasos, que definan o propongan una solución óptima al problema en cuestión.

Ha quedado claro que un algoritmo debe ser preciso, definido, finito, correcto, que debe tener al menos una salida perceptible, sencillo, legible, eficiente y eficaz. Construir un algoritmo que cumpla con las anteriores características resulta de la aplicación del análisis del problema y la verificación del propio algoritmo, para ello se han demostrado ejemplos que evidencian el cumplimiento de las instrucciones descritas.