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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
|  | RODRIGUEZ ESPINO CLAUDIA |
| *Asignatura:* | FUNDAMENTOS DE PROGRAMACIÓN |
| *Grupo:* | 1104 |
| *No de Práctica(s):* | PRÁCTICA 3 |
| *Integrante(s):* | VALENCIA MEDINA ISAC |
|  |  |
|  |  |
| *Semestre:* | PRIMER SEMESTRE 2019-1 |
| *Fecha de entrega:* | 27/AGOSTO/2018 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

GUÍA PRÁCTICA DE ESTUDIO 03: SOLUCIÒN 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.

**INTRODUCCIÓN:**

Un problema informático se puede definir como el conjunto de instancias al cual corresponde un conjunto de soluciones, junto con una relación que asocia para cada instancia del problema un subconjunto de soluciones (posiblemente vacío). Para poder solucionar un problema nos apoyamos en la Ingeniería de Software que de acuerdo a la IEEE se define como “La aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del software". Por lo que el uso y establecimiento de principios de ingeniería sólidos, son básicos para obtener un software que sea económicamente fiable y funcione eficientemente. La Ingeniería de Software provee métodos que indican cómo generar software. Estos métodos abarcan una amplia gama de tareas:

• Planeación y estimación del proyecto.

• Análisis de requerimientos del sistema y software.

• Diseño de la estructura de datos, la arquitectura del programa y el procedimiento algorítmico.

• Codificación.

• Pruebas y mantenimiento (validación y verificación)

EJERCICIO 1.

-Obtener el area de un círculo.

Datos Entrada:

-El radio **r** del círculo.

Restricciones: Ninguna

Proceso:

En primer lugar se verifica que el radio **r** introducido sea un número.

Una vez validado se emplea el valor de PI que es una constante ya definida y se multiplica por el radio al cuadrado y se almacena en la variable A.

Al final se muestra A.

Datos Salida: El valor del área del círculo A.

Algoritmo

1.Inicio

2.Se pide al usuario introduzca un número.

3.Se verifica que el dato introducido sea un número mayor a cero, en caso afirmativo se guarda el número en la variable **r** y se continua al paso 4, y en caso contrario se vuelve al paso 2.

4.Mandamos llamar el valor de PI que es una constante con un valor definido y lo multiplicamos por 2 veces el **r**, es decir, PI\*r^2 y el resultado se almacena en la variable A.

5. Se imprime en pantalla en resultado de A.

6. Fin.

Prueba de escritorio:

|  |  |  |
| --- | --- | --- |
| Valor introducido de **r** | Operación | Área del círculo |
| 3.5 | PI\*(3.5)(3.5) | 38.4845u^2 |
| -4 | PI\*(-4)(-4) |  |
| 9 | PI\*(9)(9) |  |

EJERCICIO 2:

-Solución de ecuaciones para x>2 o x<2.

EJERCICIO 3:

-Obtener las raices de una ecuación de Segundo grado.

**CONCLUSIÓN:**

El objetivo de la práctica se cumplio, pude conocer los comandos básicos para trabajar en una terminal de una distibución del Sistema operativo LINUX, conocí la importancia que tiene este Sistema como plataforma de desarrollo de Código abierto y que es la base de muchas aplicaciones y otros sistemas operativos. Fue una práctica interesante y con bastante similitud en cuanto a los comdandos que conozco del CMD de windows, diria yo que son los mismos pero con diferente sintaxis.