

Alumnos:Gonzalo Vilche, Ezequiel Acuña

Profesor: Alejandro Zeballos

1)\_ punto 2:

Etapa n1: Análisis de requisitos de prueba

Requisito funcional: Debe ingresar 3 notas.

Requisito funcional: Que saque el promedio.

Requisito no funcional: El algoritmo debe ser claro y fácil de entender y tiene que ser en php.

Etapa n2: Planificación de pruebas

Alcance: Probar el ingreso de 3 notas.

Objetivo: Qué saque el promedio.

Enfoque: Utilizar pruebas básicas de limite y de error.

Etapa n3: Diseño de casos de prueba

ID del caso	Descripción del caso de prueba	Tipo de prueba	Entrada (a, b, c)	Resultado esperado
1	El promedio es de 80 (positivo del 0 al 100)	Básico	80, 90, 70	“El promedio es de 80”
2	El promedio es de -1 (negativo)	Limite	-1, -1, -1	“error”

3	El promedio es de 1000 (positivo)	Especial	500, 1250, 1250	“error”
4	El promedio es de 100 (positivo del 0 al 100)	Básico	100, 100, 100	“El promedio es de 100”
5	El promedio es de -3 (negativo)	Limite	-2, -2, -2	“error”
6	El promedio es de 2000 (positivo)	Especial	2000, 2000, 2000	“error”

Etapa n4:

4) Utilizamos para el entorno de desarrollo una computadora que tenga instalado el programa visual studio y en PHP.

ETAPA N°5

ID del caso	Entrada (a, b, c)	Resultado Obtenido	¿paso la prueba?	observaciones
-------------	-------------------	--------------------	------------------	---------------

1	80, 90, 70	“El promedio es de 80”	si	-----
2	-1, -1, -1	“el promedio es de -1”	no	Si reconoce los números negativos
3	500, 1250, 1250	“Números muy altos”	si	-----
4	100, 100, 100	“El promedio es de 100”	si	-----
5	-2, -2, -2	“El promedio es de -2”	no	No reconoce limites negativos
6	2000, 2000, 2000	“Números muy altos”	si	----- -

2)\_ punto 3:

Etapas n1: Análisis de requisitos de prueba

Requisito funcional: Debe ingresar un radio.

Requisito funcional: Calcular y mostrar diámetro.

Requisito funcional: Calcular y mostrar el área.

Requisito funcional: Calcular y mostrar el volumen.

Requisito funcional: Calcular y mostrar la superficie de la esfera.

Requisito no funcional: El algoritmo debe ser claro y fácil de entender y tiene que ser en php.

Etapas n2: Planificación de pruebas

Alcance: Probar el ingreso de un radio.

Objetivo: Qué calcule y muestre los resultados de manera óptima.

Enfoque: Utilizar pruebas básicas de límite y de error.

Etapas n3: Diseño de casos de prueba

ID del caso	Descripción del caso de prueba	Tipo de prueba	Entrada (r)	Resultado esperado
1	radio:5 diámetro:10 perímetro:31,4 área:78,5 volumen: 523.333333333333 superficie:314 (número positivo)	Básico	5	radio:5 diámetro:10 perímetro:31,4 área:78,5 volumen: 523.333333333333 superficie:314
2	radio: diámetro: perímetro:    área: volumen: superficie: (negativo)	Límite	-5	“error”

3	radio: 500 diámetro:1000 perímetro:3140 área:758000 volumen: 523333333.33333 superficie: 3140000  (positivo)	Especial	500	radio: 500 diámetro:1000 perímetro:3140 área:758000 volumen: 523333333.33333 superficie: 3140000
4	radio:2 diámetro:4 perímetro:12,56 área:12,56 volumen:33,44 superficie:50,24 (positivo)	Básico	2	radio:2 diámetro:4 perímetro:12,56 área:12,56 volumen:33,44 superficie:50,24
5	radio: diámetro: perímetro:    área: volumen: superficie: (negativo)	Limite	-2	“error”
6	radio: diámetro: perímetro:    área: volumen: superficie: (letra)	Especial	A	“error”

Etapas n4:

4) Utilizamos para el entorno de desarrollo una computadora que tenga instalado el programa visual studio y en PHP.

ETAPA N°5

ID del caso	Entrada (R)	Resultado Obtenido	¿paso la prueba?	observaciones
1	5	radio:5 diámetro:10 perímetro:31,4 área:78,5 volumen: 523.333333333333 superficie:314	si	-----
2	-5	“No se aceptan números negativos”	si	-----
3	500	radio: 500 diámetro:1000 perímetro:3140 área:758000 volumen: 523333333.33333 superficie: 3140000	si	-----
4	2	radio:2 diámetro:4 perímetro:12,56 área:12,56 volumen:33,44 superficie:50,24	si	-----
5	-2	“No se aceptan números negativos”	si	reconoce límites negativos

6	A	“Solo acepta números”	si	----- -
---	---	-----------------------	----	------------

3)\_

## Algoritmo n°3

Etapas 1:

Requisito funcional: El algoritmo debe determinar si un número es entero.

Requisito funcional: Ver si un número es primo.

Requisito funcional: Fijarse si el número ingresado es positivo.

Requisito No funcional: Que el algoritmo sea fácil de entender.

Etapas 2:

Alcance: Probar su funcionalidad ingresando un número.

Objetivo: Determine si el número es primo y sepa identificar que sea positivo.

Enfoque: Utilizar pruebas básicas de límite y de error.

Etapas 3:

Ide del caso	Descripción del Caso de Prueba	Tipo de Prueba	Entrada (x)	Resultado Esperado
N°1	Probar que x es número primo(positivo)	Básico	2	El número 2 si es primo
N°2	Probar que x es número primo(positivo)	Básico	1	El número 1 no es primo

N°3	Probar que x es número primo(negativo)	Limite/especial	-1	El numero -1 no es positivo
N°4	Probar que x es número primo(negativo)	Limite/Especial	-4	El numero -4 no es positivo
N°5	Probar que x es número primo(positivo)	Básico/Especial	456	El numero 456 no es primo
N°6	Probar que x es número primo(negativo)	Básico/especial	-55	El número no es positivo

Etapa 4:

Para el entorno de desarrollo utilizamos una computadora, EL programa visual Studio con php, también utilizamos trazas en forma de papel.

Etapa 5:

Id del caso	Entrada (x)	Resultado obtenido	¿Paso? Si/no	Observaciones en caso de fallar
N°1	2	El numero 2 no es primo	Si	---
N°2	1	El numero 1 si es primo	Si	---
N°3	-1	El número no es positivo	Si	---
N°4	-4	El número no es positivo	Si	---
N°5	456	El numero 456 no es primo	Si	---
N°6	-55	El número no es positivo	si	---