

UNIVERSIDAD ESTATAL A DISTANCIA  
ESCUELA CIENCIAS EXACTAS Y  
NATURALES

PROGRAMA DE INFORMÁTICA

Tarea 3

Programa de salud física y consumo de calorías

Nombre del Curso: Programación intermedia

Grupo No. 1

Nombre Completo: Isaac Morera Vargas

Número de cédula: 116200798

Sede San jose

25 de febrero.

I CUATRIMESTRE 2018

## **Tabla de contenido.**

Introducción. Pag 3

Desarrollo. Pag 4

Conclusiones. Pag 5

Recomendaciones. Pag 5

Bibliografía. Pag 6.

## Introducción.

Se analiza el siguiente caso:

Es usted una persona apasionada del deporte y la salud física; al mismo tiempo es estudiante de la carrera de Ingeniería en Informática en una de las universidades con mayor población estudiantil del país. Usualmente usted realiza un deporte varias veces a la semana y en un momento de creatividad tuvo la genial idea de poner su conocimiento de programación en JAVA a su servicio. Por tanto, comenzó a crear una aplicación de escritorio donde usted puede llevar el control de su entrenamiento y peso corporal.

- Aplicar los conceptos de Herencia y Polimorfismo para crear una nueva clase llamada Bicicleta que pueda heredar de la clase Abs.
- Realizar los cálculos correctos para el ejercicio con bicicleta.
- Crear los métodos necesarios para los nuevos cálculos.
- Mejorar la GUI.
- El programa deberá tener una GUI

Analizando lo que se pide meticulosamente, se llega a la conclusión de que para resolver este caso generalmente se necesitarán los conocimientos sobre POO, clases, métodos y subclases, esta vez el programa deberá evolucionar una vez mas así como su estructura para poder complementar una GUI mejorada una vez mas mientras mantiene las funcionalidades hasta el momento, además se espera que el código adapte el temario de herencia y polimorfismo ya que se espera poder darle a la clase Bicicleta los atributos de la clase Abs además de poder manipular de una forma concisa el temario visto hasta el momento. Ya que como objetivos a resolver nos piden la creación de clases y métodos como solución anudados a una interfaz, así como decisiones booleanas, con un funcionamiento incluso diferente en el manejo de excepciones.

## Desarrollo.

La metodología a seguir es matemática/lógica mas estructuralmente, todas al mismo tiempo ya que la solución del caso requiere de un uso de la lógica bien estructurado porque el lenguaje de programación lo exige para funcionar a nivel de sintaxis aparte de las ya exigencias del proyecto, sin olvidar la indispensable parte matemática que siempre es aplicada ya que el problema planteado exige el llenado y la aplicación de formulas.

El desarrollo teórico teniendo en cuenta la metodología, es concluir antes de iniciar las necesidades lógicas para lograr desarrollar el problema, las cuales tiene que ver con las decisiones sobre cuando utilizar las clases, cuando y como desarrollar los métodos anidados en los bloques que evitarían errores de procesamiento y cuales instrucciones de bucles pueden ser útiles sin abusar de ellas como se tiende normalmente.

Se nos pide la creación de clases para la resolución del problema:

Principal: Sera la clase encargada del proceso principal del programa(**main**) ya que sera quien posea el switch del menu de opciones, según la elección decidí hacer el uso de funciones para el menú ya que es simple y eficaz, dicho menú lo desplegué como sub-clases private para que solo se pueda acceder desde el menú y no cause errores por la interacción desde otras ubicaciones. Así como la inclusión de un ciclo para que solo se escape introduciendo la opción adecuada.

Persona: Sera la clase encargada del proceso para obtener los datos necesario para calcular el indice de masa corporal, pero al mismo tiempo según lo que se nos pide, calcular este indice debe ser un método, por lo tanto dentro de la clase Persona cree los constructores para el método y el método Persona para que coincida con la clase y cubrir la parte del calculo.

ABS: Sera la clase encargada del proceso para obtener los datos necesario para calcular el consumo de calorías, pero al mismo tiempo según lo que se nos pide, calcular este indice debe ser un método, por lo tanto dentro de la clase ABS cree los constructores para el método y el método ABS para que coincida con la clase y cubrir la parte del calculo.

Bicicleta: Esta clase debe heredar sus propiedades y calculos de la clase Abs en la medida de lo posible, con un pequeno analisis podemos entender que lo que se puede heredar de Abs, serian sus ventanas, sus botones, y sus metodos para los calculos

Una de las ultimas partes que se nos pide es utilizar catch and try para evitar los errores, dentro de los cuales pude identificar problemas a nivel de usuario, estos problemas son la introducción incorrecta de números o letras donde debería ir un numero provocando un error fatal a nivel de sistema o incluso la inclusión de 0 ya que es matemáticamente imposible utilizarlo para las divisiones, las cuales están presentes en la resolución del problema.

Ahora se analizara la creación de la interfaz del programa o GUI mejorada ya que al analizar la forma en la que funciona esta parte de la programación en Java nos llegamos a dar cuenta que el programa presentara una pequena modificacion de la GUI anterior debido a la clase Bicicleta ya que los formularios funcionan de una forma diferente, lo único que el programa mantuvo intacto es su código para calcular resultados y mostrar estos hasta cierto punto.

Ademas se esperara que el estudiantes cree archivos de texto para guardar todos estos resultados que lance los programas.

## **Conclusión.**

La tarea represento un reto en su solución, se presentaron ciertos problemas, mejor conocidos como problemas de sintaxis ya que los conceptos de métodos y jerarquía de clases son nuevos para este nivel, por lo tanto su dominio y aplicación toma tiempo y esfuerzo real, el proceso matemático/lógico como siempre fue desafiante, pero la resolución también fue satisfactoria ya que se alcanzo satisfactoriamente lo pedido en el trabajo, fue especialmente difícil comprender la el uso de poliformismo y herencia como aplicarla y sus alcances ya que las restricciones de funcionamiento son realmente específicas, pero las soluciones de las clases principal, persona, abs y bibicleta fueron satisfactorias ya que produjeron los resultados esperados y pedidos en las instrucciones.

Así como la creación de modificacion GUI, presentando el comportamiento esperado ya que la reestructuración que presento fue importante y la forma en que se obtienen los datos en esta, además se tuvo que ser realmente cuidadoso a la hora de crear los try, catch, ya que el usuario puede introducir datos o no seguir el orden adecuado y esto produciría errores en los resultados o resultados sin ningún sentido.

También se dio la solución requerida a la creación de archivos T.X.T que exigen los nuevos requerimientos para el proyecto final, así como el uso adecuado de la herencia y poliformismo que se esperaba obtener de las clases Abs-Bicileta.

Esperando que en el futuro se puedan desarrollar satisfactoriamente los trabajos venideros porque se tenga un manejo adecuado de los conocimientos y como siempre, nunca es suficiente cuando se trata de mejorar el razonamiento matemático/lógico ya que la efectividad y eficacia del problema viene de estos dos razonamiento fundamentales en el mundo de las computadoras.

## **Recomendaciones.**

Actualmente no se si es por la falta de experiencia sobre el tema, pero hasta el momento en los capítulos estudiados no vi claramente la explicación sobre el direccionamiento de funciones o métodos a nivel de switch o a otra paginas dentro de Java para el tema, que es complicado y difícil de asimilar, el cual la tarea lo pide pero el libro no es expresamente claro, por lo tanto es especialmente necesario prestar atención a este tema estudiarlo mucho mas al fondo para entender su alcance y restricciones, e ir mas allá del libro para comprender con ejemplos a nivel practico como funcionan porque unicamente a nivel teórico es imposible.

## **Bibliografía.**

Melvin Nunez (2016). Programación intermedia en Java. Primera edición. Deitel&Associates.

Melvin Nunez (2010). Introducción a la programación orientada a objetos. Primera edición. Deitel&Associates.

Melvin Nunez Viquez(2011). Introducción a la programación orientada a objetos II. Primera edición. Deitel&Associates.

Juan David Meza Gonzales(2012-2018). Objetos, clases y constructores en Java. Crear una clase y un objeto. Class y new en Java, [en línea]. Disponible en:  
<https://www.programarya.com/Cursos/Java/Objetos-y-Clases>