

[Todos mis módulos](#) / [Mis módulos](#) / [2021-2022](#) / [Desarrollo de aplicaciones multiplataforma](#) / [Programación](#) / Estructuras de control (10%)

/ [PROG03 Tarea de evaluación 01. Realiza un programa en Java \(40%\)](#)

Descripción

[Entrega](#)

[Editar](#)

[Ver entrega](#)

PROG03 Tarea de evaluación 01. Realiza un programa en Java (40%)

Límite de entrega: viernes, 19 de noviembre de 2021, 23:59

Número máximo de ficheros: 1

Tipo de trabajo: Individual

PROG03 TE01 REALIZA UN PROGRAMA EN JAVA (40%)

ENUNCIADO

Esta tarea nos va a permitir analizar las diferencias entre un programa bien estructurado y otro que no lo está.

Para ello, seguiremos las indicaciones de los siguientes vídeos.

Vídeos

En el primero de ellos, analizaremos una versión mal estructurada de un programa para calcular el índice de masa corporal de una persona. Fijaos que aunque el programa calcula lo que se pide, su comprensión y mantenimiento resultan difíciles y no podemos aprovechar los resultados obtenidos.

[Versión mal estructurada](#)

En el siguiente, estudiaremos una versión bien estructurada. Los métodos realizan tareas definidas, sus resultados pueden ser reutilizados y además ayudan a que el programa se entienda y se pueda mantener mejor.

Ejemplo de versión bien estructurada

Ficheros .java utilizados: [BuenIMC.java](#) y [MalIMC.java](#)

Una vez finalizados, realizaremos el siguiente programa en el que tendremos que utilizar **estructuras condicionales IF-ELSE, mientras seguimos practicando el uso de variables, bucles FOR y métodos con paso de parámetros y sentencia return** así como, las funcionalidades de las clases de las librerías de java.

El programa permitirá **calcular la nota media de un estudiante** a partir de la calificación del examen final, el examen parcial y las tareas realizadas durante el curso.

Para ello, primero mostrará la siguiente **introducción al programa**:

```
Este programa lee las calificaciones de exámenes y tareas
y calcula la nota final del módulo para un estudiante.
También podría hacerlo para 2 estudiantes y comparar sus notas. Funcionaría sin problemas
```

A continuación, pedirá toda la información necesaria para **calcular la nota parcial** y mostrará tanto la nota final como la nota final ponderada tal y como se muestra a continuación. Pedirá el peso que tiene el parcial en la nota final, la nota sobre 100 que obtenido el estudiante y si tiene puntos extra o no. Los puntos extra se sumarán a la nota del examen hasta llegar a 100 que será el valor máximo para este concepto. Es decir, si alguien tiene una nota de 95 y 10 puntos extra, el valor que obtendrá será 100 y no 105 que sería lo que realmente le correspondería. La nota final ponderada la calculará multiplicando la nota por el peso entre 100:

```
Parcial:
Introduce el peso del examen (0-100): 25
Introduce la calificación del examen (0-100): 70
¿Has obtenido puntos extra (1=Si, 2=No)? 1
Introduce el total de puntos extra: 5
Nota final = 75 / 100
Nota final ponderada = 18,8 / 25
```

Seguidamente, pedirá toda la información necesaria para **calcular la nota final**. El funcionamiento será idéntico al del parcial:

```
Final:
Introduce el peso del examen (0-100): 40
Introduce la calificación del examen (0-100): 75
¿Has obtenido puntos extra (1=Si, 2=No)? 2
Nota final = 75 / 100
Nota final ponderada = 30,0 / 40
```

Por último, pedirá toda la información necesaria para **calcular la nota de las tareas** realizadas durante el curso y mostrará tanto la nota final como la nota final ponderada tal y como se muestra a continuación. Pedirá el peso que tienen las tareas en la nota final y el número de tareas que ha realizado. Por cada una de esas tareas, pedirá la nota obtenida y la nota máxima que se podía obtener. También pedirá el número de presenciales a las que se ha asistido. Cada una de ellas valdrá 5 puntos hasta un máximo de 30. La nota final será la suma de las notas obtenidas tanto en las tareas como en las presenciales sobre el total de puntos posibles que será la suma de las puntuaciones máximas. en este caso la nota es 100, 40 + 50 + 10, sobre un total de 160, 60 + 70 + 30. La nota ponderada, se obtendrá multiplicando la nota por el peso entre la puntuación máxima. En este caso, 100 * 35 / 160:

```
Tareas:
Introduce el peso de las tareas (0-100): 35
Introduce el número de tareas: 2
Tarea 1: Introduce la puntuación obtenida y la puntuación máxima: 40 60
Tarea 2: Introduce la puntuación obtenida y la puntuación máxima: 50 70
Introduce el número de presenciales a las que has atendido: 2
Nota de las presenciales = 10 / 30
Nota final = 100 / 160
Nota final ponderada = 21,9 / 35
```

Una vez obtenidas estas 3 notas, calculará la nota final como la suma de las 3 notas ponderadas, 18,8 + 30,0 + 21,9, y la mostrará por pantalla. Además, convertirá el valor obtenido a la siguiente escala de 0 a 4 y dará el mensaje correspondiente:

Si la nota es igual o superior a 85, se le dará un valor de 3.0 en la escala y se visualizará el mensaje "Excelente trabajo".
Si la nota está entre 84.99 y 75, se le dará un valor de 2.0 en la escala y se visualizará el mensaje "Gran trabajo".
Si la nota está entre 74.99 y 60, se le dará un valor de 0.7 en la escala y se visualizará el mensaje "Todavía tienes trabajo por hacer".
Si la nota es inferior a 60, se le dará un valor de 0.0 en la escala y se visualizará el mensaje "Otra vez será".

```
La calificación total obtenida es: 70,6
La nota final en una escala del 0 al 4 es al menos: 0.7
Todavía tienes trabajo por hacer
```

RECURSOS

Para realizar esta tarea puedes utilizar jGRASP o el editor VPL de Moodle que además te dirá si la salida del programa coincide con la propuesta.

INDICACIONES DE DESARROLLO

Realiza el programa por partes. Escribe el código poco a poco compilándolo y testeándolo frecuentemente.

Utiliza métodos, parámetros y returns para estructurar el programa y evitar redundancias. Para conseguir la máxima nota, se utilizarán al menos 4 métodos con parámetros y al menos uno de ellos usará la sentencia return.
La estructura utilizada te debería permitir calcular la media de 2 alumnos diferentes y compararlas
Como en el programa anterior en el main no tendremos ningún println.
Existirá un único objeto de tipo Scanner que se creará en el main. Se podrá utilizar en cualquier otro punto del programa si se pasa como parámetro.

Para calcular la nota final de las tareas se necesitará utilizar el algoritmo acumulador.

Un vez conseguidos los valores, fíjate en el formato exigido:

Todos los valores se muestran con un único decimal. Utiliza el método [System.out.printf\(\)](#) para formatearlo de la siguiente manera.

```
double num = 1.2345;
System.out.printf("El número redondeado a un decimal es %.1f.\n", num);
```

El valor de num se escribe sustituyendo a %.1f. La f indica el tipo de datos double y el .1 el número de dígitos después del punto.

Elige los tipos de datos de manera adecuada. Se penalizará utilizar un tipo double cuando los datos sean de tipo int.

En el programa se utilizan valores de control. Al menos uno de ellos debería ser una constante.

Da nombres significativos a métodos, variables y parámetros siguiendo además las reglas de Java. Utiliza correctamente la indentación y los espacios en blanco. Incluye cabeceras explicativas al principio del programa y de cada uno de los métodos, así como junto a cualquier línea de código que lo pueda necesitar. En la cabecera de los métodos, incluye los parámetros que utiliza y se devuelve algún dato.

INDICACIONES DE ENTREGA

El programa se escribirá directamente en el IDE proporcionado o se copiará en él una vez finalizado.

En la cabecera, además de la información que describe el programa, se añadirá el enlace a la evaluación realizada, un enlace a Google Drive o a Youtube según el formato elegido

CRITERIOS DE CALIFICACIÓN. TOTAL 10 PUNTOS

La tarea se evaluará siguiendo la siguiente rúbrica:

Funcionalidad (70%).

Comentarios (15%).

Legibilidad (15%).

Con ella, se calculará la nota de vuestro programa y se comparará con la que habéis indicado en la autoevaluación.

Si la diferencia es menor de un punto, se calculará la nota media entre las dos.

Si la diferencia es mayor, se utilizará directamente la calificación de la profesora.

En los casos en los que falte la autoevaluación, se utilizará la calificación de la profesora con una penalización de un punto.

[VPL](#)

Navega por la unidad

◀ PROG03 Tarea de aprendizaje 02. Uso de IF-ELSE

Ir a...

PROG03 Tarea de aprendizaje 03. While ▶

Contacta con nosotros:

Dirección: Paseo de Ubarburu 39, Edificio EnerTIC of. 206 · Donostia San Sebastián

Telefono : 945 567 953

E-mail: info@birt.eus

Twitter: @Birt_LH



