

PRÁCTICA CALCULADORA

Objetivos

- Fomentar el trabajo en equipo.
- Fomentar el uso de herramientas colaborativas.
- Aprender a usar las herramientas de trabajo colaborativo Github y de forma opcional Trello/Github Project Boards.
- Aprender a gestionar correctamente flujos de trabajo colaborativo con git.
- Poner en práctica todo lo aprendido hasta ahora sobre la programación estructura.

Agrupamientos

- Mínimo 4 alumnos, máximo 7 alumnos.
- En este primer trabajo en grupo los alumnos pueden establecer sus propios grupos siempre y cuando no haya alumnos sin grupo y se cumpla el número mínimo y máximo de alumnos por grupo.
- Si hubiese problemas para hacer todos los grupos, el profesor configurará la distribución de los grupos intentando que sean lo más compensados posible.

Descripción de la práctica

Escribir un programa implemente las funciones básicas de una calculadora. El programa mostrará el siguiente menú principal:

```
*****
**  CALCULADORA  **
*****
1. Operaciones aritméticas
2. Conversiones de base
-----
0. Salir del programa
```

Elija una opción:

1) **Operaciones aritméticas.** Al elegir esta opción mostrará el siguiente submenú:

```
*****
**  OPERACIONES ARITMÉTICAS  **
*****
1. Sumar
2. Restar
3. Multiplicar
4. Dividir
5. Resto de la división
```

0. Volver al menú principal

Elija una opción:

1. **Sumar.** Solicitará dos números al usuario y mostrará la suma de los números introducidos.
2. **Restar.** Solicitará dos números al usuario (el primero será el minuendo y el segundo el sustraendo) y mostrará la resta de los números introducidos.
3. **Multiplicar.** Solicitará dos números al usuario (multiplicando y multiplicador), y mostrará la multiplicación de los números introducidos.
4. **Dividir.** Solicitará dos números al usuario (dividendo y divisor) y mostrará la división de los números introducidos.
5. **Resto de la división.** Solicitará dos números al usuario (dividendo y divisor) y mostrará el resto de la división de los números introducidos.
0. **Volver al menú principal.** Al pulsar 0 volverá al menú principal de la calculadora.

2) **Conversiones de base.** Al elegir esta opción mostrará el siguiente submenú:

```
*****  
** CONVERSIONES DE BASE **  
*****  
1. De decimal a binario  
2. De decimal a hexadecimal  
3. De binario a decimal  
4. De binario a hexadecimal  
5. De hexadecimal a binario  
6. De hexadecimal a decimal  
-----  
0. Volver al menú principal
```

Elija una opción:

Para realizar este apartado no se pueden utilizar métodos ya existentes que realicen directamente la conversión, tales como Integer.toBinaryString(), Integer.toHexString(), String.format("%0X", decimal) o similares. Deben programarse cada uno de los pasos para realizar los cambios de base.

1. **De decimal a binario.** Solicitará el número decimal al usuario y calculará su equivalente en binario.
2. **De decimal a hexadecimal.** Solicitará el número decimal al usuario y calculará su equivalente en hexadecimal.
3. **De binario a decimal.** Solicitará el número binario al usuario y calculará su equivalente en decimal.
4. **De binario a hexadecimal.** Solicitará el número binario al usuario y calculará su equivalente en hexadecimal.
5. **De hexadecimal a binario.** Solicitará el número hexadecimal al usuario y calculará su equivalente en binario.
6. **De hexadecimal a decimal.** Solicitará el número hexadecimal al usuario y calculará su equivalente en decimal.
0. **Volver al menú principal.** Al pulsar 0 volverá al menú principal de la calculadora.

0) **Salir del programa.** Finalizará la ejecución de la calculadora.

Requisitos

- Todos los alumnos deben tener una cuenta de Github (<https://github.com>)
- Uno de los alumnos del grupo creará el repositorio y añadirá al profesor y al resto de los alumnos del grupo como colaboradores.
- Los alumnos irán realizando sus aportaciones mediante Pull Requests. Para que un alumno pueda ser evaluado, deberá haber realizado al menos 3 commits.
- El profesor realizará, sin previo aviso antes de la fecha de finalización de la práctica, al menos un Pull Request en cada grupo. El objetivo de este Pull Request, es fomentar el buen hábito de la sincronización del proyecto antes de cada sesión de trabajo.
- Después del Pull Request del profesor, todos los alumnos deberán aportar al menos un commit más.
- De forma opcional, se realizará un kanban donde se llevará un registro de las tareas que se van proponiendo (To-do), a quien se asignan y su estado de avance (In-progress) y cuando finalizan (Done).

Plazo de entrega

- La fecha límite de finalización de la práctica es el lunes 27/11/2023 (incluido)
- El repositorio no puede ser modificado después de la fecha indicada.

Criterios de evaluación

- **Criterios de evaluación generales.**
 - Nivel de descripción de los mensajes de cada commit (1 punto)
 - Correcta gestión de los menús (1,5 puntos)
 - Validación de la entrada de datos del usuario (cuando sea necesaria) (1 punto)
 - Funcionamiento de las operaciones aritméticas (1,5 puntos)
 - Funcionamiento de los cambios de base (2 puntos)
 - Estructura del programa y adecuación de los métodos siguiendo convenientemente los principios de divide y vencerás, reutilización de código y facilidad de mantenimiento (3 puntos)
 - **Penalizaciones**
 - Tabulación/indentación incorrectas (-0.2 por cada línea mal tabulada hasta un máximo de -1 punto).
 - No utilizar comentarios javadoc en los métodos (-0.2 por cada método no comentado hasta un máximo de -1 punto).
 - Mal uso de los convenios de escritura para código Java (-0.2 por cada identificador mal escrito hasta un máximo de -1 punto).
- **Criterios de evaluación individuales.**

Aunque se trata de un trabajo en grupo, la nota final de cada alumno vendrá determinada por las aportaciones que haya realizado. La nota general del grupo se multiplicará por el número de alumnos, y luego según el nivel de aportación se obtendrá un porcentaje para cada alumno. Los aspectos que se valorarán para determinar su nivel de aportación son:

 - Participación del alumno en la práctica (cantidad de commits y líneas de código por commit). (40% a repartir entre los miembros del grupo)

- Los parámetros de medida para este apartado son:
 - Cantidad de commits
 - Líneas de código por commit.
- Nivel de corrección del código final aportado por el alumno (60% a repartir entre los miembros del grupo)
 - Los parámetros de medida para este apartado son:
 - Nomenclatura y uso correcto de constantes y variables.
 - Nomenclatura y uso correcto de métodos.
 - Aplicación correcta de técnicas de diseño basadas

Por ejemplo, en un grupo de 4 alumnos en el que la nota general sea de 7 puntos, la nota final de un alumno con una participación del 15% sobre 40% y un nivel de corrección del 15% sobre 60% sería de:

$7 \text{ puntos} \times 4 \text{ alumnos} = 28 \text{ puntos totales}$

$28 \times 15\% = 4,2 \text{ puntos de participación}$

$28 \times 15\% = 3,75 \text{ puntos de nivel de corrección}$

Nota final del alumno = $4,2 + 3,75 = \mathbf{7,95 \text{ puntos.}}$

- **Puntuación adicional (+1 punto)**

- Todos los alumnos del grupo pueden conseguir un punto extra en la práctica si, junto con la documentación del proyecto, aportan el tablero kanban con al menos 3 paneles (To-do, In-progress y Done) donde se vea quien crea las tareas y a quien son asignadas. Los 2 tableros kanban aceptados para esta práctica son:
 - Trello
 - Github Project Board