

## EVIDENCIA EJEMPLO DE UN EJERCICIO DE PROGRAMACIÓN

Escribir un programa que calcule el perímetro y el área de un círculo, a partir del valor del radio proporcionado por el usuario.

### CÓDIGO DEL PROGRAMA

```
// Armando Esteban Quito Blanco
// Programación y Métodos Numéricos
// Ejercicio 0
// 15-FEB-2017
disp("Cálculo de perímetro y área de un círculo")
radio = input("Radio en centímetros? ");
perimetro = 2*pi*radio;
area = pi*radio**2;
disp("El perímetro es "+string(perimetro)+" cm")
disp("El área es "+string(area)+" cm^2")
```

El enunciado del ejercicio es opcional, pero recomendable.

El programa debe incluir, como comentarios iniciales, nombre del alumno, materia, fecha y número de ejercicio.

Usar un tipo de letra de espaciado fijo, por ejemplo:  
Courier New  
Lucida Console

### CORRIDA DE PRUEBA

```
-->exec("Circulo.sce");

Cálculo de perímetro y área de un círculo
Radio en centímetros? 7.5
El perímetro es 47.12389 cm
El área es 176.71459 cm^2
```

Incluir la corrida de prueba. Si el ejercicio pide generar gráficas, incluirlas también.

Los comentarios respecto al ejercicio, qué aprendieron, si tuvieron alguna dificultad... (tipo "conclusiones").

### COMENTARIOS

Éste es el primer programa "funcional" que realizamos, aunque se fue modificando varias veces a lo largo de la clase. Comenzó simplemente con las mismas instrucciones que se habían probado previamente en la consola de Scilab, para calcular perímetro y área de un círculo de 5 cm de radio. En el transcurso de la clase, se agregó algo de funcionalidad al programa, primero con la instrucción `input` que permitió al usuario especificar el radio (en vez de que siempre fuera 5 cm), y luego con la adición de las instrucciones `disp` para mostrar el resultado mediante mensajes más "amigables" con el usuario.

También se hizo énfasis en la importancia de incluir el punto y coma al final de las instrucciones del programa donde se calculan las variables `perimetro` y `area`, para que no se muestren estos cálculos intermedios en la pantalla, así como el punto y coma después de la instrucción `exec` para correr el programa sin que se muestren de nuevo las instrucciones del programa a medida que se ejecuta.

Aunque es un programa sencillo, este ejemplo nos permitió aprender los pasos básicos para escribir y ejecutar un programa en Scilab, así como la manera en la que se prepararán los reportes de los ejercicios para el portafolio de evidencias.

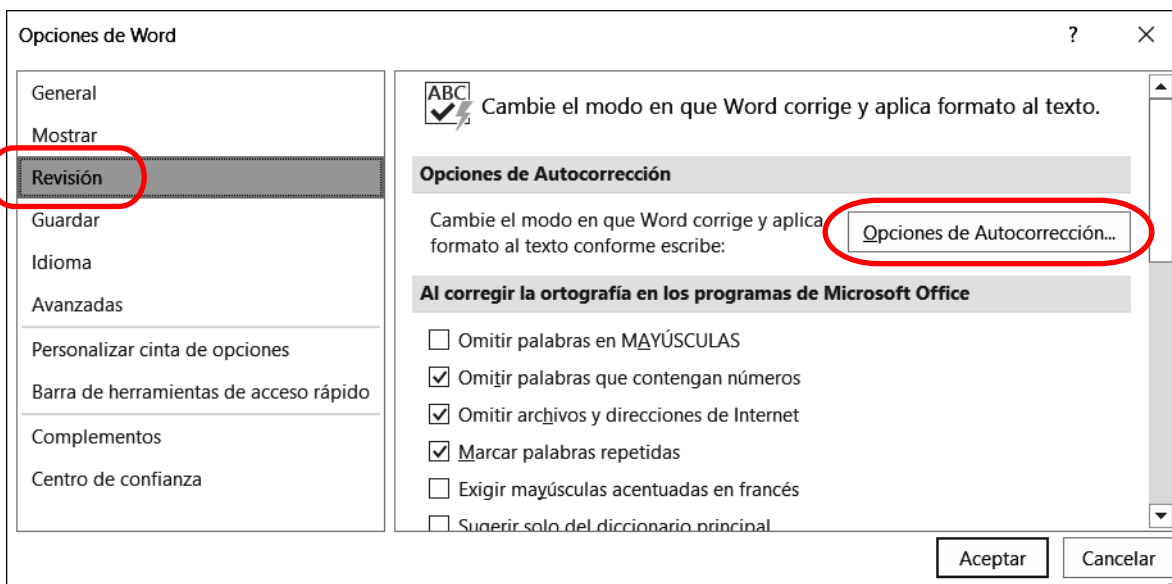
Las hojas deben llevar nombre y materia en la parte inferior.

## ¿Cómo desactivar comillas tipográficas al pegar código en Word?

Estas indicaciones son para la versión 2016 de Microsoft Word. Otras versiones pueden requerir un procedimiento diferente.

De forma predeterminada, Microsoft Word cambia las comillas rectas ( " " ) por comillas tipográficas ( “ ” ) cuando se copia y pega un texto desde otro programa. En muchos lenguajes de programación, las comillas rectas tienen la función particular de encerrar cadenas de texto (por ejemplo "Hola mundo!" ), por lo que el cambio a comillas tipográficas causa errores en los programas.

Para desactivar en Word el cambio automático de comillas, abrir el menú **Archivo** y seleccionar **Opciones**. Aparece una ventana como la siguiente, en la que hay que seleccionar en el lado izquierdo **Revisión**, y a continuación en el lado derecho el botón **Opciones de Autocorrección**.



En la ventana que aparece, seleccionar la pestaña Autoformato, y desactivar la casilla de opción "Comillas rectas" con "comillas tipográficas".

