

**Nombre:** Luis Fernando González Chávez      **No. de Matrícula.:** ZAP408

**Materia:** Fundamentos de la Programación Grupo: Dev 22-1 Turno: Matutino

**Carrera:** Ingeniería en Desarrollo de Software Interactivo y Videojuegos

**Tema:** Entrada y Salida de Datos **No:** R.1 11

**Fecha propuesta:** 30/Sep/2021      **Fecha de Entrega:** 30/Sep/2021

**Escuela:** Instituto Universitario Amerike      **Plantel** Guadalajara

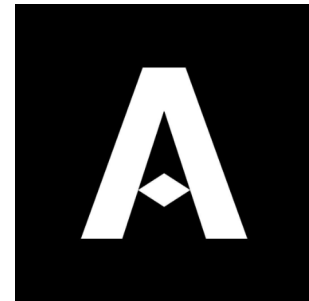
**Calle:** C. Montemorelos      **No:** 3503      **Colonia:** Rinconada de la Calma      **C.P.:** 45080

**Teléfono:** 3336326100

**Ciudad:** Zapopan



Ferchus



## Firma del alumno (a)

## Firma de revisión fecha

Qué se evalúa:	10 pts.	7 pts.	4 pts.	
Entrega electrónica	Es en tiempo y forma al iniciar la clase. (1 pts.)	Después de 30 minutos de iniciada la clase. (.7 pts.)	Al minuto 40. (Posteriormente ya no se reciben). (.4pts.)	
Del formato.	Cumple con todos los elementos solicitados. (1 pts.)	No cumple con dos elementos solicitados. (.7 pts.)	No cumple con tres o más elementos solicitados. (.4pts.)	
La ortografía.	Tiene dos errores ortográficos. (1 pts.)	Tiene de tres a cuatro errores ortográficos. (.7 pts.)	Tiene cinco o más errores ortográficos. (.4pts.)	
Del tema y objetivo.	La teoría y ejemplos corresponden al tema tratado. (1 pts.)	La teoría o ejemplos no corresponden al tema tratado. (.7 pts.)	La teoría y ejemplos no corresponden al tema tratado. (.4pts.)	
El programa y los cálculos.	Los parámetros y componentes corresponden al 100% de lo planeado. (1 pts.)	El programa arroja un error o componente no corresponden al 100% de lo planeado. (.7 pts.)	El programa arroja dos errores o componentes no corresponden al 100% de lo calculado. (.4pts.)	
Diagramas.	Los diagramas a bloques, de flujo y esquemáticos son acorde al de la práctica y siguen una secuencia lógica. (1 pts.)	Los diagramas a bloques, o de flujo o esquemáticos no son acorde al de la práctica y o no siguen una secuencia lógica. (.7 pts.)	Los diagramas a bloques, de flujo y esquemáticos no son acorde al de la práctica y o no siguen una secuencia lógica. (.4pts.)	
La tabla de valores.	Los valores calculados y medidos presentan una desviación máxima del 10%. (1 pts.)	Los valores calculados y medidos presentan una desviación máxima del 15%. (.7 pts.)	Los valores calculados y medidos presentan una desviación máxima del 20%. (.4pts.)	
Las observaciones y conclusiones.	Son específicas y congruentes con la práctica. (1 pts.)	Las observaciones o conclusiones son específicas y congruentes con la práctica. (.7 pts.)	Las observaciones y las conclusiones no son específicas y congruentes con la práctica. (.4pts.)	
Bibliografía.	Es acorde al (los) tema (s) tratado (s) y está completa (1 pts.)	Es acorde a algún (os) tema (s) tratado (s), le falta algún elemento que la conforman (.7 pts.)	No es acorde al (los) tema (s) tratado (s), le faltan 2 elementos que la conforma (.4pts.)	
Fuentes de consulta.	Es acorde al (los) tema (s) tratado (s) (1 pts.)	Es acorde a algún (os) tema (s) tratado (s) (.7 pts.)	Es acorde a algún (los) tema (s) tratado (s) (.4pts.)	

**Nombre:** Luis Fernando González Chávez

**Tema:** Entrada y Salida de Datos

**No.** T-11

**Página** 1

## Índice

Teoría 3

scanf 3

printf 3

cin 3

cout 3

Cálculos 4

Diagramas 4

Tabla (comparativa) 4

Repositorio 6

Bibliografía 7

Fuentes de consulta 7

## Teoría

Como tenemos que usar `cin`, `cout`, `scanf` y `printf` para guardar e imprimir los tipos de datos `int`, `float`, `char`, `double` y `string`, primero debemos de declarar las variables. En este caso, hay un tipo de variable que no puedes usar sin agregar una librería, `string`. Para usarlo, escribiremos `"#include <string>"`. Después de esto puedes usarla como cualquier otro tipo de variable. También necesitamos usar `"#include <stdio.h>"` para poder usar `scanf` y `printf`.

scanf: Función que pide una entrada de valor por parte del usuario. Para usarlo, debes saber qué tipo de variable estás leyendo y utilizar el formato correspondiente (pueden encontrarlos en [https://en.wikipedia.org/wiki/Scanf\\_format\\_string](https://en.wikipedia.org/wiki/Scanf_format_string)). Los parámetros son:

`scanf("formato", &nombre variable)` *si es una string o la variable es un puntero en sí, no escribes "&"*.

En versiones recientes de Visual Studio, debes usar `scanf_s` si quieres usar la configuración predeterminada del compilador, pues no es muy segura. Para usar `scanf`, debes escribir una excepción en la configuración del compilador.

printf: Función que toma un formato string y argumentos opcionales y los muestra al usuario. Puedes mostrar un mensaje simple sin variables, o puedes incluir en tu programa. Los parámetros son:

`printf("mensaje con el formato de las variables donde quieras que aparezcan", nombres de variables en orden).`

cin: Otra función que pide una entrada de valor por parte del usuario. Este es mucho más fácil de usar, pues no requieres saber qué tipo de variables estás leyendo, lo único que necesitas es el nombre de la variable. Tiene varias derivadas (por ejemplo, `cin.get()`) pero la función principal se utiliza con los símbolos `>>` para concatenar la función.

Ejemplo: `cin >> nombre_variable;`

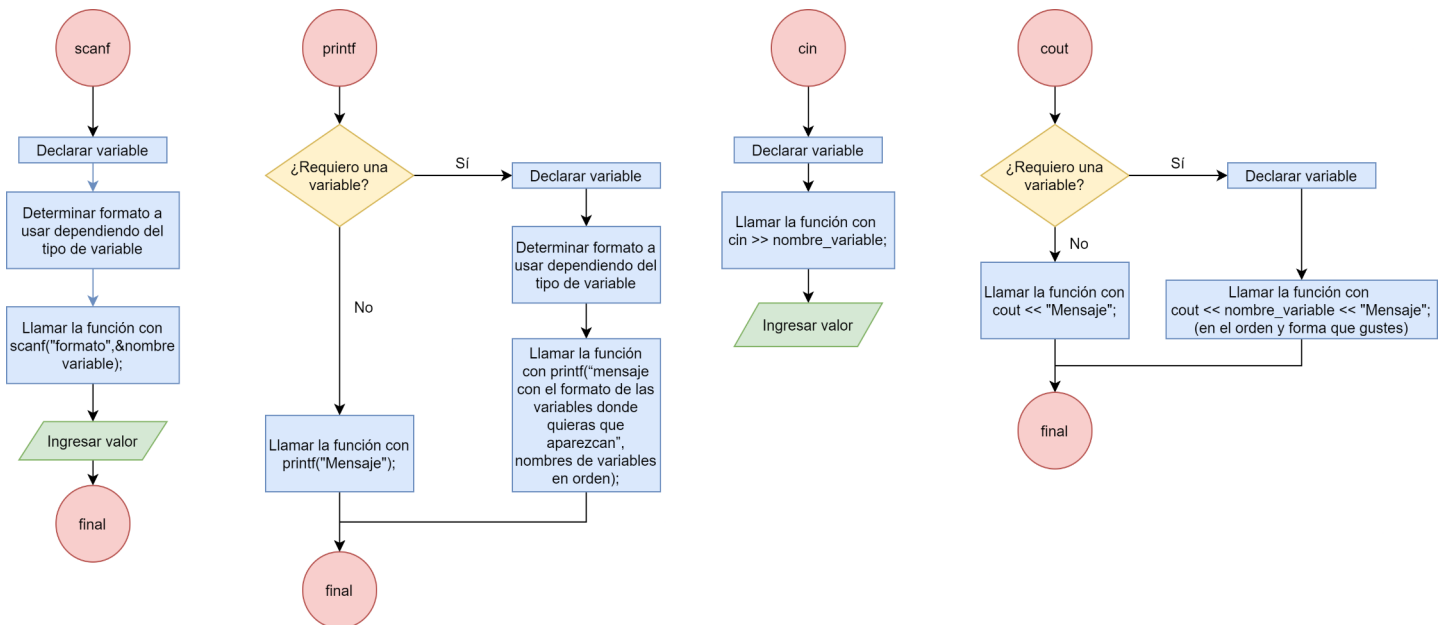
cout: Otra función que le muestra información al usuario. Al igual que `cin`, es mucho más fácil de usar que `printf`, pues solo necesitas el nombre de la variable. Puedes utilizar `endl` para terminar la línea.

Ejemplo `cout << nombre_variable << endl;`

## Cálculos

n/a

## Diagramas



## Tabla (comparativa)

Función	Expectativa	Resultado	Conclusión
scanf / printf (int)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
scanf / printf (float)	Mostrar valor ingresado de acorde	Mostrar valor ingresado de acorde	Se ejecutó de manera correcta.

	al tipo.	al tipo.	
scanf / printf (double)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
scanf / printf (char)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
scanf / printf (string)	Mostrar valor ingresado de acorde al tipo.	Problemas al compilar y al correr. Me dice que debe ser un const char*.	scanf y printf no funcionan bien con las strings de la librería <string>, requieren strings en la forma de const char*.
scanf / printf (char[])	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta y uno debe de usar este tipo de variables para mostrar texto dentro de estas funciones.
cin / cout (int)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
cin / cout (float)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
cin / cout (double)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
cin / cout (char)	Mostrar valor ingresado de acorde al tipo.	Mostrar valor ingresado de acorde al tipo.	Se ejecutó de manera correcta.
cin / cout (string)	Mostrar valor	Mostrar valor	Se ejecutó de manera

ingresado de acorde  
al tipo.

ingresado de acorde  
al tipo.

correcta.

*Nota.* Usé scanf\_s para no cambiar la configuración del compilador.

## Repositorio

[https://github.com/Zap408/ZAP408\\_FundamentosProgramacion\\_Practica4](https://github.com/Zap408/ZAP408_FundamentosProgramacion_Practica4)

## Bibliografía

Stroustrup, B. (2013). 2.2 The Basics. En *The C++ Programming Language* (4th ed., p. 38). Addison-Wesley Professional.

## Fuentes de consulta

cplusplus.com. (s. f.-a). (float.h) - C++ Reference. Recuperado 30 de septiembre de 2021, de <https://www.cplusplus.com/reference/cfloat/>

cplusplus.com. (s. f.-b). (limits.h) - C++ Reference. Recuperado 30 de septiembre de 2021, de <https://www.cplusplus.com/reference/climits/>

Microsoft. (2019, 21 octubre). scanf, \_scanf\_l, wscanf, \_wscanf\_l. Microsoft Docs. <https://docs.microsoft.com/en-us/cpp/c-runtime-library/reference/scanf-scanf-l-wscanf-wscanf-l?view=msvc-160>

Microsoft. (2020, 26 octubre). Format Specification Syntax: `printf` and `wprintf` Functions. Microsoft Docs. <https://docs.microsoft.com/en-us/cpp/c-runtime-library/format-specification-syntax-printf-and-wprintf-functions?view=msvc-160>

Painless Programming. (2019, 11 enero). Error Handling with cin, integer overflow. YouTube. <https://www.youtube.com/watch?v=XRLlk7HyR3s>

tutorialspoint. (s. f.). C library function - scanf(). Www.Tutorialspoint.Com. Recuperado 30 de septiembre de 2021, de

[https://www.tutorialspoint.com/c\\_standard\\_library/c\\_function\\_scanf.htm](https://www.tutorialspoint.com/c_standard_library/c_function_scanf.htm)

Wikipedia contributors. (2021, 15 febrero). Scanf format string. Wikipedia.

[https://en.wikipedia.org/wiki/Scanf\\_format\\_string](https://en.wikipedia.org/wiki/Scanf_format_string)