

Es necesario cumplir con los siguientes requerimientos para que los ejercicios puedan ser evaluados:

- *Limitarse a codificar lo indispensable para cumplir con lo que en la descripción de cada ejercicio se indica. Esto es, que no se solicite leer de teclado algún otro dato que, para el programa no haya sido solicitado, como un nombre de alumno, código, etc.*
- *No es necesario programar un menú de opciones para los ejercicios que no lo soliciten, tampoco preocuparse por agregarle una “interfaz” con impresiones de cadenas como “marquitos”, por ejemplo:*

```
*****  
Ejercicio n  
Alumno: Nombre de Alumno  
*****
```

- *Evitar encerrar en bucles iterativos la ejecución del programa.*
- *Mostrar las impresiones de los datos calculados, en líneas independientes o bien separadas por un espacio, cuidando que el dato numérico se imprima correctamente, por ejemplo:*

En un programa que se soliciten las medidas de una figura para el cálculo del área, la impresión puede ser de la siguiente manera. Nota: luego de la entrada de los datos, hay un salto de línea.

```
Ingresa base: 10  
Ingresa altura: 6
```

El área del rectángulo es: 60

- *Es indispensable en la entrega de cada programa, que el código fuente sea completamente en idioma español (a excepción de lo correspondiente a la API del lenguaje).*
- *Para los ejercicios del 0 al 40, es indispensable codificar todo el programa haciendo uso de solo una subrutina (esto es, escribir todas las instrucciones del programa dentro de la subrutina `main()`); y, para estos ejercicios de menor complejidad, no incluir archivos de cabecera `.h` creados por el alumno.*

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

Al final de cada ejercicio imprimir los valores de cada variable (solo si sí es requerido utilizar variables), e imprimirlas en el orden descrito en cada ejercicio.

0. Implementar un programa que imprima en pantalla en líneas separadas de la pantalla: su NickName, carrera, preparatoria o bachillerato de procedencia, sus motivos para cursar la carrera y, la lista de todos los lenguajes de programación que conoce, aunque sea un poco (en cuyo caso entregar un archivo fuente de cada lenguaje de programación y que en todos los lenguajes sea el mismo programa que se solicita aquí), o bien, decir "ninguno" o "ninguno mas que lo poco de este".
1. Implemente un programa que calcule el área de un triángulo usando variables `int` nombradas "a", "b" y "h". No es necesario solicitar datos al usuario.
 2. Implemente un programa que calcule el área de un triángulo usando variables `int` nombradas "areaTriangulo", "base" y "altura". No es necesario solicitar datos al usuario.
 3. Implemente un programa que calcule el área de un círculo usando variables `float` nombradas "a", "r" y "pi". Imprimir los valores de cada variable en el orden descrito y considerar $\pi=3.1416$. No es necesario solicitar datos al usuario.
 4. Implemente un programa que calcule el área de un círculo usando variables `float` nombradas "areaCirculo", "radio" y "pi". Imprimir los valores de cada variable en el orden descrito y considerar $\pi=3.1416$. No es necesario solicitar datos al usuario.
 5. Implemente un programa que calcule el área de un círculo usando variables `float` nombradas "areaCirculo", "radio" y constante `#define PI 3.1416`. No es necesario solicitar datos al usuario.
 6. Implemente un programa que, usando `scanf()` para base y altura, calcule el área de un triángulo usando variables `int` nombradas "areaTriangulo", "base" y "altura".
 7. Implemente un programa que usando `scanf()` para radio, calcule el área de un círculo usando variables `float` nombradas "areaCirculo", "radio" y constante `#define PI 3.1416`.

8. Escriba un programa que solicite al usuario dos valores enteros a y b (deseablemente enteros) y realice las siguientes operaciones:

- $a + b$
- $a - b$
- $b - a$
- $a * b$
- a / b
- b / a
- $a \% b$
- $b \% a$

El programa debe **mostrar los valores leídos** y el **resultado** de cada operación realizada como se sugiere en el siguiente ejemplo:

```
Ingresa un valor para a: 10
Ingresa un valor para b: 5
```

```
El valor de a es: 10
El valor de b es: 5
```

```
a + b = 15
a - b = 5
b - a = -5
a * b = 50
a / b = 2
b / a = 0
a % b = 0
b % a = 5
```

Para los siguientes los programas, se soliciten al usuario los datos de entrada necesarios para los cálculos.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

9. Escriba un programa que haga la conversión de grados Celsius a Fahrenheit mediante la fórmula:

$$F = 1.8 \times ^\circ C + 32$$

Debe pedir el valor a convertir y **mostrarlo en pantalla junto con su valor equivalente calculado.**

10. Escriba un programa que haga la conversión de grados Fahrenheit a Celsius mediante los siguientes dos métodos:

- Haciendo un cálculo aproximado como sigue
 - o Restar 32 al valor leído para adaptar el equivalente en la escala Fahrenheit.
 - o Dividir el valor obtenido entre 2 (equivalente a multiplicar por 0.5).
 - o Tomar 1/10 del valor generado y sumarlo al valor obtenido anteriormente, lo cual daría la aproximación en grados.
 - o Por ejemplo, para convertir 60 F a $^\circ C$ se hace:

$$60 - 32 = 28 \quad \frac{28}{2} = 14 \quad \frac{14}{10} = 1.414 + 1.4 \approx 15.4^\circ C$$

- Haciendo una conversión exacta usando la siguiente fórmula

$$^\circ C = \frac{5}{9}(F - 32) \quad \text{O bien} \quad ^\circ C = \frac{F - 32}{1.8}$$

$$\frac{5}{9}(60 - 32) = 15.55^\circ C \quad \frac{60 - 32}{1.8} = 15.55^\circ C$$

Se deben imprimir el valor leído y el valor obtenido en cada método y comparar en ejecución los resultados de ambos métodos para saber la diferencia que hay del valor aproximado al valor exacto e imprimirla. Para el ejemplo, la diferencia a imprimir de ambos métodos es:

$$15.4 - 15.55 = -0.15$$

Para los siguientes los programas, se soliciten al usuario los datos de entrada necesarios para los cálculos.

11. Implemente el intercambio de valores de tres valores a, b y c en un programa mediante la siguiente manera:

- a = valor inicial de c
- b = valor inicial de a
- c = valor inicial de b

El programa debe imprimir el valor original de las 3 variables, luego hacer el intercambio y finalmente imprimir los valores de las 3 variables.

Nota: se necesita una variable auxiliar para realizar los intercambios y no perder el valor inicial de los valores. Usar en el programa solo 4 variables. Para toda impresión de variable (a, b, c) se imprima la variable mencionada y no otra variable, por ejemplo:

En C

```
printf("Valor de a es %d\n", a);  
printf("Valor de b es %d\n", b);  
printf("Valor de c es %d\n", c);
```

En C++

```
cout << "Valor de a es " << a << endl;  
cout << "Valor de b es " << b << endl;  
cout << "Valor de c es " << c << endl;
```

Nunca hacer algo como

```
printf("Valor de a es %d\n", c); /* disque imprime la  
a, pero es el valor de c el que se está mostrando*/  
  
cout << "Valor de a es " << c << endl;
```

Los ejercicios siguientes requieren cumplir los Requerimientos de Valor Agregado en Código Fuente, hasta la M.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

12. Implemente un programa que, dada una constante `MAXIMO_ASISTENCIAS = 34`, **calcule el porcentaje de asistencias** de un alumno si falta X veces a clase. **Solicitar al usuario la cantidad de faltas** y el programa imprima el porcentaje de asistencias.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

13. Escriba un programa que obtenga del usuario los valores para un arreglo de enteros, de 10 celdas dicho arreglo. Luego, capture un factor (un multiplicador). Después, cada elemento del vector debe ser multiplicado por el factor y depositar los resultados en un segundo arreglo. Finalmente, después lo anterior (que haya multiplicado todas y cada una de las celdas), debe ser mostrado el vector resultante (**cada valor, impreso después de un salto de línea**). Si conoce las estructuras de control iterativas, no las aplique en este ejercicio.
14. Implementar y usar un registro llamado "Alumno" con 3 campos de información: un campo de tipo cadena, otro de tipo entero y otro de tipo flotante. Inicializar los campos a partir de entrada de consola **en el orden descrito** y, al final del programa, imprimirlos siguiendo la misma organización. Alumnos del curso de Programación (primer semestre) pueden entregar en C++ de ser necesario (por si hay dificultad para entregar en lenguaje C).
15. Implementar y usar en un mismo programa un registro "Arbol" con 3 campos (uno de tipo entero, otro de tipo carácter y uno más de tipo cadena) y otro registro "Libro", con 3 campos de información (cadena, entero y flotante). Inicializar los campos a partir de entrada de consola y al final del programa imprimir todos los campos de ambos registros en el orden solicitado. Alumnos del curso de Programación (primer semestre) pueden entregar en C++ de ser necesario (por si hay dificultad para entregar en lenguaje C).

16. Evolucionar el ejercicio 14, añadiendo un registro "Carrera" con los campos "idCarrera" (int) y "nombre" (cadena) y colocar al registro "Alumno" el campo "idCarrera".

Solicitar primero los datos para la carrera y luego los del alumno; asignar el valor del ID de la carrera, al campo "idCarrera" del alumno.

Cuidar que el orden de los datos a solicitar sea el mismo que se describe en el ejercicio.

Estudiantes del curso de Programación pueden entregar en C++ de ser necesario (por si hay dificultad para entregar en lenguaje C).

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

17. En una tienda se venden artículos de primera necesidad, a los cuales se les aplica un descuento del 20% de la compra total, si esta es igual o mayor a \$50. Escriba un programa que, a partir del importe total de la compra muestre lo que debe pagar el cliente; informar en pantalla solo lo que debe pagar el cliente a partir del monto de compra, sin decirle al usuario nada más que eso. Usar solo el "if disparador" (if sin else). Usar solo una variable. **Usar solo un "if" en todo el programa.**

18. Escriba un programa que al recibir como datos tres valores enteros diferentes entre sí, determine si los mismos están en orden creciente; informar en pantalla una y solo una de las siguientes dos salidas: "están en orden creciente" o "no están en orden creciente". Tips:

1, 2, 3, 4, 5	<u>"están en orden creciente"</u>
1, 3, 4, 5, 10	<u>"están en orden creciente"</u>
1, 1, 2, 3, 4	<u>"están en orden no decreciente"</u> y también <u>"no están en orden creciente"</u>
1, 1, 2, 2, 3	<u>"están en orden no decreciente"</u> y también <u>"no están en orden creciente"</u>

19. Escriba un programa que, al recibir como dato el salario de un profesor de una universidad, calcule su incremento de acuerdo con el siguiente criterio e imprima en pantalla **el nuevo salario**, o bien **el salario tal cual lo tenía** si no le aplica incremento.

Salario < 18,000	incremento del 12%
18,000 <= salario <= 30,000	incremento del 8%
30,000 < salario <= 50,000	incremento del 7%

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

20. Escriba un programa que pida el mes del año (1 a 12) y el día del mes (1 a 31) y, en base a los valores introducidos, imprima con letra qué mes es y qué estación del año es.

Por ejemplo, si se leyeran los valores 3 y 25, el programa imprimirá:

El mes es marzo y la estación es primavera

Así mismo, si se leyeran los valores 3 y 20, el programa imprimirá:

El mes es marzo y la estación es invierno

Nota: la ortografía correcta para las estaciones y meses del año en español, se escriben en minúscula; sólo se inician con mayúscula cuando forman parte de nombres que exigen la escritura de sus componentes con mayúscula inicial, como ocurre con los nombres de festividades, fechas o acontecimientos históricos, vías urbanas, edificios, etc.

21. Implementar un programa que solicite al usuario el valor para una variable "anio" y que resuelva si se trata de un año bisiesto o no, dado lo siguiente:

- Un año bisiesto se presenta cada 4 años contados a partir del año cero, esto es: 0, 4, 8, 12...etc. son años bisiestos.
- Hay una excepción para los años bisiestos, esto es cada 100 años: 100, 200, 300, 500, 600, 700...etc. no son años bisiestos.
- Cómo se habrá notado en la serie anterior, hay una excepción a la excepción, esto es que, cada 400 años (400, 800, 1200...etc.), sí son años bisiestos.

Tip: si la expresión `anio%4` calcula un 0, anio es un múltiplo de 4.

Mostrar en pantalla el año leído y su correcta clasificación tal como se indica a continuación:

1200 es año bisiesto

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

22. Implemente un programa que solicite dos valores numéricos y la operación aritmética básica (suma, resta, etc.) a realizar por medio de su símbolo. El programa debe imprimir el resultado de la operación y los valores leídos. Pueden ser valores enteros o reales. Use la estructura `switch` para determinar la operación.

Un ejemplo de la ejecución del programa puede ser como se muestra a continuación:

Ingresa el primer número: 10

Ingresa el segundo número: 2

Elige la operación a realizar (+, -, * o /): +

El resultado de $10 + 2 = 12$

23. Escriba un programa que haga el cálculo del área de una de las siguientes figuras **en base a un menú de opciones** para círculo (opción 1), cuadrado (opción 2), triángulo (opción 3) o rectángulo (opción 4).

El programa debe pedir los datos necesarios para obtener el área. Use la estructura de control switch para el menú de opciones, sin preguntar si se desea ejecutar de nuevo el programa cuando se termine de ejecutar todo el procedimiento. Imprimir la figura y su resultado.

Elige opción: 1

Ingresa radio: 9

El área del círculo es: 254.4696

24. Realice un programa que calcule el importe a pagar por un vehículo al circular por una autopista. El vehículo puede ser motocicleta, automóvil, camión (utilizar un menú para elegir el tipo de vehículo). Use la estructura switch para el selector, cuidando el orden que se describe a continuación. El importe se calculará de la siguiente manera:

- Las motos pagarán \$35 + IVA
- Los automóviles pagarán \$48.50 + IVA
- Los camiones pagarán \$57 + \$18 por cada tonelada

Mostrar en pantalla el tipo de vehículo y el importe a pagar. Considerar IVA en 16%.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

25. Escriba un programa que solicite un número "x". El programa utilice una sola variable, de modo que todo cómputo (suma, resta, multiplicación, división, potencia, valor absoluto, etc.) asigne el resultado de nuevo en "x". Si "x" es positivo lo eleve al cuadrado, luego, si el nuevo "x" es par le sume 5, de lo contrario le reste 5. Si "x" es negativo lo eleve al cubo, luego, para el nuevo "x" determinar su valor absoluto originando un nuevo "x", finalmente, si este último "x" es mayor a 100 le sume 100, de lo contrario le reste 100. Imprimir el valor final de "x". Ejemplos:

- Si $x=10$, imprima al final $x=105$
- Si $x=9$, imprima $x=76$
- Si $x=-5$, imprima $x=225$
- Si $x=-4$, imprima $x=-36$
- Si $x=0$, imprima $x=5$

Para este ejercicio considerar al 0 como positivo.

26. Implemente un programa que solicite al usuario el tiempo inicial (hora, minuto, segundo) y el tiempo final (hora, minuto, segundo) y calcule la diferencia de tiempo en horas, minutos y segundos.

El programa debe mostrar el tiempo transcurrido en formato HH:MM:SS (horas:minutos:segundos) o informarle al usuario si este capturó un tiempo final con valor inferior al tiempo inicial. Asumir que el usuario capturará la hora correctamente en el rango 0 a 59 segundos, 0 y 59 minutos y 0 a 23 horas. Al imprimir la diferencia de tiempo no imprimir más de 59 segundos ni más de 59 minutos. Ejemplos de captura:

- a. Hora inicial 0:10:20, Hora final 0:20:30, se imprima como resultado "Diferencia 0:10:10"
- b. Hora inicial 0:10:58, Hora final 0:20:30, se imprima como resultado "Diferencia 0:9:32"
- c. Hora inicial 0:10:20, Hora final 0:9:59, se imprima como resultado "Capturé un tiempo final inferior al tiempo inicial"

Es necesario realizar un scanf individual para capturar cada uno de los 6 datos necesarios: hora inicial, minuto inicial, segundo inicial, hora final, minuto final, segundo final; es decir, 6 scanf en total.

27. Implementar un programa que pida al usuario que escriba el operador (+, -, *, /, %) de su elección y luego solicite los valores para unas variables "a" y "b", finalmente se escribirá la operación realizada y su resultado.

Escribe el operador (+, -, *, /, %): %

Dame el valor de a: 20

Dame el valor de b: 3

20 % 3 = 2

Entregar en el comprimido 2 carpetas con el código fuente utilizando, para la selección canónica, una estructura de control selectiva diferente y cuyo nombre de carpeta sea la estructura de control, una carpeta para "if" y otra para "switch".

Los ejercicios en adelante requieren de cumplir los Requerimientos de Valor Agregado en Código Fuente hasta el requerimiento Z.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

28. Un encuestador realiza una consulta en la calle a 100 personas para saber qué prefieren de lo siguiente:

- a. Los políticos deben bajarse el sueldo y dejar de robar.
- b. Los políticos deben conservar su sueldo y dejar de robar.
- c. Es indistinto, ya que los políticos siempre roban.

Escriba un programa que obtenga la opinión de cada persona e imprima al final cuántas personas votaron por cada una de las opciones. Utilice la estructura de control do-while. Solicite de la consola escribir las letras "a", "b", "c" (contemplando recibir tanto mayúsculas como minúsculas) y luego la tecla "entrar".

Al final, los resultados debieran mostrar la cuenta de cada respuesta, por ejemplo:

```
34 personas votaron a)
29 personas votaron b)
38 personas votaron c)
```

29. Mostrar en pantalla el abecedario (al menos el abecedario del inglés), mostrando el par mayúscula y minúscula Aa, Bb, Cc, Dd, etc. Utilice la estructura de control do-while. Ver el ejemplo en carpeta "Ejemplos PRO\01 Estructuras de Informacion Simples\01 Variables y Constantes\15 Variables caracter". No es necesaria entrada de datos.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

30. Escriba un programa que solicite 50 valores reales e imprima al final lo siguiente:

- Cuántos son positivos y cuántos son negativos
- Cuántos son pares y cuántos son impares
- La suma de los valores positivos y la suma de los valores negativos
- La suma de los valores pares y la suma de los valores impares
- El promedio de todos los valores introducidos.

Utilice la estructura de control do-while. El 0 sea considerado positivo. Para determinar si son pares o impares considere solo la parte entera de los valores capturados. Un ejemplo de presentación de resultados es el siguiente:

```
Pares: 36
Impares: 14
Positivos: 44
Negativos: 6
Suma de Pares = 592.00
Suma de Impares = 416.00
Suma de Positivos = 1158.00
Suma de Negativos = -150
```

31. Escriba un programa que pregunte a cuántas personas se desea saludar y que imprima lo siguiente (para el caso por ejemplo de 2 personas):

```
Hola persona #1, te saludo  
Hola persona #2, te saludo
```

En caso de capturar un 0 imprima: "no hay nadie".

- Entregar 2 fuentes de este programa en el mismo comprimido: uno que utilice `while` (ubicado en una subcarpeta "while") y otro que utilice `for` (ubicado en una subcarpeta "for", ambas subcarpetas dentro de la carpeta que corresponda al lenguaje de programación utilizado). [Escribir "Tiempo=" para el primer programa elaborado \(ya sea que primero lo haga con while y luego con for, o viceversa\)](#), y ["TiempoTraduccion=" para cuando utilicen otra estructura de control para este mismo ejercicio.](#)

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

32. Escriba un programa que calcule el promedio de las estaturas (deseablemente en metros) y edades de un grupo de N personas; se requiere leer el valor de N y luego, para cada persona, leer su estatura y edad.

Considere validar que N puede ser cero. El programa imprima los promedios de las estaturas y las edades.

- Entregar 2 fuentes de este programa en el mismo comprimido: uno que utilice `while` (ubicado en una subcarpeta "while") y otro que utilice `for` (ubicado en una subcarpeta "for", ambas subcarpetas dentro de la carpeta que corresponda al lenguaje de programación utilizado). [Escribir "Tiempo=" para el primer programa elaborado \(ya sea que primero lo haga con while y luego con for, o viceversa\)](#), y ["TiempoTraduccion=" para cuando utilicen otra estructura de control para este mismo ejercicio.](#)

33. Un médico desea conocer el peso promedio de sus pacientes en base a la categoría que pertenecen:

- a. Niños: 0 a 12 años
- b. Adolescentes: 13 a 17 años.
- c. Jóvenes: 18 a 29 años.
- d. Adultos jóvenes: 30 a 59 años.
- e. Adultos mayores: 60 en adelante.

Escriba un programa que pida la cantidad de pacientes del médico y para cada uno de ellos solicite la edad y peso. Al final imprima el peso promedio en cada una de las categorías que maneja. Nótese que puede no haber pacientes en alguna categoría y valide para evitar un fallo.

- Entregar 2 fuentes de este programa en el mismo comprimido: uno que utilice `while` (ubicado en una subcarpeta "while") y otro que utilice `for` (ubicado en una subcarpeta "for", ambas subcarpetas dentro de la carpeta que corresponda al lenguaje de programación utilizado). [Escribir "Tiempo=" para el primer programa elaborado \(ya sea que primero lo haga con `while` y luego con `for`, o viceversa\)](#), y ["TiempoTraduccion="](#) para cuando utilicen otra estructura de control para este mismo ejercicio.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

34. Implementar un programa que pida al usuario que ingrese los valores para el multiplicando y el último multiplicador y, posteriormente, se imprima la tabla de multiplicar correspondiente, empezando desde el 1, seguido de una pausa.

Validar el caso donde para el multiplicador se capture un valor menor o igual a cero, y se muestre un mensaje al respecto.

- Entregar 3 carpetas en el mismo comprimido, con el código fuente utilizado para imprimir la tabla una estructura de control iterativa diferente, nombradas según sea la estructura:
 - a. "do-while"
 - b. "while"
 - c. "for"

...sendas subcarpetas dentro de la carpeta que corresponda al lenguaje de programación utilizado)

Las 3 versiones deben imprimir en consola exactamente igual ante el usuario.

```
Dame el multiplicando = 7
Dime hasta cual multiplicador = 10
Tabla de multiplicar del 7...
7      X      1      =      7
7      X      2      =     14
7      X      3      =     21
7      X      4      =     28
7      X      5      =     35
7      X      6      =     42
7      X      7      =     49
7      X      8      =     56
7      X      9      =     63
7      X     10      =     70
```

Escribir "Tiempo=" para el primer programa elaborado (ya sea que primero lo haga con do-while, o while o for), y "TiempoTraduccion=" para cuando utilicen otra estructura de control para este mismo ejercicio.

35. En una escuela secundaria se hace un censo para conocer:

- Cuántos alumnos son mujeres y cuántos son hombres
- La cantidad de personas que tienen 12 o 13, 14 o 15 o más de 15 años cumplidos.

Implemente un programa que solicite la cantidad de alumnos de la secundaria y, para cada uno de ellos, se indique si es hombre (con el carácter 'h' o 'H') o mujer (usando el carácter 'm' o 'M') y su edad. Dicho programa debe imprimir los resultados buscados en el censo como se ejemplifica a continuación:

Cantidad de hombres: n
Cantidad de mujeres: n
Personas de 12 o 13: n
Personas de 14 o 15: n
Personas de más de 15: n

- Entregar 2 fuentes de este programa en el mismo comprimido: uno que utilice `while` (ubicado en una subcarpeta "while") y otro que utilice `for` (ubicado en una subcarpeta "for", ambas subcarpetas dentro de la carpeta que corresponda al lenguaje de programación utilizado). Escribir "Tiempo=" para el primer programa elaborado (ya sea que primero lo haga con `while` y luego con `for`, o viceversa), y "TiempoTraduccion=" para cuando utilicen otra estructura de control para este mismo ejercicio.

Los ejercicios en adelante requieren de cumplir los Requerimientos de Valor Agregado en Código Fuente (hasta el requerimiento Z):

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

36. Escriba un programa que permita al usuario la captura de 8 valores en un arreglo de enteros del mismo tamaño, utilizando para ello una estructura de control iterativa. Después de capturar todos los valores, enseguida se sumen los elementos de todas las celdas y, para esto, será necesario usar un acumulador que vaya almacenando la sumatoria. Aplicar una estructura de control iterativa diferente a la usada para la captura. Finalmente, el programa debe mostrar en pantalla la suma total, es decir: se debe imprimir el acumulador. Por estructuras diferentes, sea respecto a usar para una iteración `do-while`, para otra `while`, y para otra `for`.

37. Escriba un programa que obtenga del usuario los valores para un arreglo de enteros, de 10 celdas dicho arreglo, utilizando una estructura de control iterativa. Luego, al terminar la iterativa anterior, capture un multiplicador. Luego, cada elemento del vector debe ser multiplicado por el multiplicador, depositando el resultado en el mismo vector, usando una estructura de control iterativa diferente a la primera. Finalmente, después de terminar la iteración anterior, sea mostrado el vector resultante, usando una estructura de control iterativa diferente a las 2 anteriores. Por estructuras diferentes, sea respecto a usar para una iteración do-while, para otra while, y para otra for.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

38. Escriba un bucle que calcule la suma de cada tercer entero, comenzando por $i=2$ (es decir, calculando la suma de $2+5+8+11+\dots$) para todos los valores de i menores que 100. Imprimir solo el resultado de la suma.
39. Escriba un programa que solicite un valor entero y haga lo siguiente:
- Si el número es positivo encuentre todas las sumas de números enteros positivos consecutivos hacia la derecha (orden ascendente) que den el número introducido, por ejemplo: $50=8+9+10+11+12$ y $50=11+12+13+14$
 - Si el número es negativo, éste encuentre todas las sumas de número enteros negativos consecutivos hacia la izquierda (orden descendente) que den el número introducido por ejemplo: $-50=-8-9-10-11-12$ y $-50=-11-12-13-14$
 - Si no existe ninguna suma imprima que no es posible obtenerla.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

40. Codifique un programa que solicite un valor entero positivo k y en base a ese valor realice lo siguiente:
- Escriba los números desde 1 hasta el valor k , cada uno tantas veces como indique su propio valor
 - Calcule la suma y la multiplicación de ellos.

Por ejemplo, si $k=4$, debe obtenerse la salida:

$1 = 1, 1 = 1$
 $2 + 2 = 4, 2 \times 2 = 4$
 $3 + 3 + 3 = 9, 3 \times 3 \times 3 = 27$
 $4 + 4 + 4 + 4 = 16, 4 \times 4 \times 4 \times 4 = 256$

41. Implemente y use una subrutina sin parámetros, que muestre el menú de opciones para el ejercicio 23, donde la última opción sea “salir”; el programa cumpla con el ejercicio 23 y que muestre cíclicamente el menú hasta que el usuario elija “salir”.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

42. Implemente una subrutina que muestre una presentación gráfica en modo texto (*googlea* “text art”), que abarque toda la pantalla, luego espere por la tecla entrar; usar dicho procedimiento como complemento de cualquier programa de los previamente solicitados en este documento, llamando al procedimiento al principio del programa principal. La presentación no exceda 24 líneas impresas más otra línea para una “pausa” para continuar.
43. Implementar y usar desde `main()` una función que reciba un entero y devuelva su factorial, en caso de parámetro negativo, devolver un -1.
44. Implemente una función que reciba base tipo real y exponente entero y que devuelva la potencia contemplando inclusive exponente negativo; usar dicha función desde el programa principal; el programa incluya SOLO las librerías básicas.

Los ejercicios en adelante requieren de cumplir los Requerimientos de Valor Agregado en Código Fuente (hasta el requerimiento FF):

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

45. Implementar una función que reciba un arreglo de enteros y **devuelva el menor valor** que encuentre. Usar dicha función desde el programa principal con un vector de al menos 4 celdas.
- Ingresar desde consola el tamaño del arreglo y enseguida los valores que contendrá la colección de enteros. Imprimir sólo el menor valor encontrado.
46. Implementar una función que reciba un arreglo de enteros y **devuelva verdadero** si todos los números son diferentes, en otro caso **devuelva falso**. Usar dicha función desde el programa principal con un vector de al menos 4 celdas.
- Ingresar desde consola el tamaño del arreglo y enseguida los valores que contendrá la colección de enteros. Imprimir verdadero o falso, según corresponda.

Los ejercicios en adelante requieren de cumplir los Requerimientos de Valor Agregado en Código Fuente (hasta el requerimiento KK).

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

En adelante, entiéndase por "programa ABC", aquél que gestione un listado de registros y cuente con las opciones de menú: "Alta", "Baja", "Consulta" (de un registro), "Cambio" (de datos de un registro), "Listado" (de todos los registros) y "Búsqueda" (de un registro particular o varios que coincidan con un determinado patrón de búsqueda).

47. Después de haber estudiado bien el programa de la carpeta "Ejemplos PRO\09 Estructuras de Control Compuestas\07 El ABC, Altas, Bajas, Consultas y Cambios", implementar desde cero un programa ABC que funcione de manera similar al ejemplo, solo que usando registros cuyo nombre sea Alumno ó, Arbol ó, Libro, con atributos propios del tipo de dato elegido, arreglo de registros, programación modular y optativamente la opción de menú "búsqueda". Evitar lo más posible estar consultando muy seguido cómo está hecho el ejemplo. Queda por cuenta del estudiante el entrenar hasta lograr implementar un programa así, desde cero, en menos de 1 hora, programando tiempo corrido.

48. Modificar el programa ABC del ejercicio 47, para cumplir con todo lo solicitado, pero removiendo el campo "libre" del registro modelado y, representando el estado "libre" de los registros en un arreglo de *booleanos*, por separado del arreglo de registros. Esto es, si el arreglo de *booleanos*, en su celda 0 indica verdadero (valor 1), entonces es posible capturar un registro nuevo en la celda 0 del arreglo de registros, de lo contrario entiéndase que el registro tiene ya datos útiles para consultarlos y listarlos. Procurar de ser posible elaborar la solución evitando recurrir a consultar la solución en carpeta "Ejemplos PRO\09 Estructuras de Control Compuestas\08 El ABC arreglo de booleanos"; al final comparar su solución contra la del ejemplo dado.
49. Después de haber estudiado bien el programa de la carpeta "Ejemplos PRO\09 Estructuras de Control Compuestas\09 El ABC recorriendo registros", implementar desde cero un programa ABC que funcione de manera similar al ejemplo, solo que usando registros cuyo nombre sea Alumno ó, Arbol ó, Libro (y que el tipo de dato estructurado sea diferente al elegido por el estudiante para el ejercicio anterior), con atributos propios del tipo de dato elegido, arreglo de registros, programación modular y que incluya en el menú optativamente la opción "búsqueda". Evitar lo más posible estar consultando muy seguido cómo está hecho el ejemplo.
50. Después de haber estudiado bien el programa de la carpeta "Ejemplos PRO\09 Estructuras de Control Compuestas\10 Reemplazando registro", implementar desde cero un programa ABC que funcione de manera similar al ejemplo, solo que usando registros cuyo nombre sea Alumno, Arbol o Libro (y que el tipo de dato estructurado sea diferente al elegido para los dos ejercicios anteriores), con atributos propios del tipo de dato elegido, arreglo de registros, programación modular y que incluya en el menú optativamente la opción "búsqueda". Evitar lo más posible estar consultando muy seguido cómo está hecho el ejemplo.

51. Implementar un programa ABC, para gestionar un listado de materias con sus prerequisites, aplicando la técnica de “recorrer registros eliminados”; si se capturara el siguiente listado de materias con sus prerequisites (máximo 3 por materia), al solicitar del menú la opción “Listar registros”, se imprimía en consola tal cual se muestra a continuación (nótese que varias materias muestran más de un prerequisite).

- Incluir en el programa una **subrutina de inicialización**, la cual emule la carga de información proveniente de un archivo y que dicha subrutina se ejecute antes de que se muestre el menú principal; en lugar de cargar información de un archivo, se agregue en la lista de materias, todos y cada uno de los registros citados en la ilustración previa, usando valores *hardcodeados*.

Clave	Nombre	Prerequisite(s)
I5882	Programación	(Lógica)
		(Español)
I5892	Matemática Discreta	(Aritmética)
		(Álgebra)
		(Trigonometría)
I5886	Estructuras de Datos I	I5882
		I5892
I5888	Estructuras de Datos II	I5886
I5890	Bases de Datos	I5886
I5898	Ingeniería de Software I	I5890
I5909	Programación para Internet	I5890
I7029	Sistemas Operativos	I5888
I7035	Sistemas Concurrentes y Distribuidos	I7029
		I5898

52. Elabore un programa que multiplique una matriz $m \in R^{5 \times 5}$ entre un vector $v \in R^5$. Inicializando los datos desde entrada de consola. Mostrar únicamente el vector resultante del producto.

53. Implemente un programa que calcule la multiplicación matricial de 2 matrices $m_1, m_2 \in R^{3 \times 3}$ inicializadas con valores secuenciales.

- Pedir un valor inicial al usuario y, a partir de dicho valor, inicializar cada celda de ambas matrices con valores diferentes. Imprimir las tres matrices resultantes.

54. Escriba un programa que llene una matriz $m \in R^{4 \times 5}$ con valores secuenciales, comenzando desde un valor ingresado por el usuario y luego hacer lo siguiente:

- Suma cada columna de la matriz y almacene los resultados en un vector
- Suma cada fila de la matriz y almacene los resultados en otro vector.
- Imprima la matriz y a sus lados los vectores de suma.

Matriz					
1	2	3	4	5	15
6	7	8	9	10	40
11	12	13	14	15	65
16	17	18	19	20	90
					} Vector Fila
34	38	42	46	50	
Vector Columna					

55. Escriba un programa que solicite valores reales para una matriz $m \in R^{4 \times 4}$ y muestre lo siguiente:

- El contenido de la matriz.
 - La suma de los elementos de la matriz
 - El promedio de los elementos de la matriz.
 - La posición $p_{i,j}$ en la matriz del elemento menor y el valor de dicho elemento:
[i, j] = x
 - La posición $p_{i,j}$ en la matriz del elemento mayor y el valor de dicho elemento:
[i, j] = x
 - La cantidad de elementos que son iguales al elemento menor.
 - La cantidad de elementos que son iguales al elemento mayor.
 - La cantidad de elementos que son mayores al elemento menor.
 - La cantidad de elementos que son menores al elemento mayor.
- Ejercer la programación modular