



**PROGRAMACIÓN 1 (CC47)**  
**CICLO 2019-01**  
**Examen Parcial**  
**Tema A**

**Profesora:** Todos  
**Duración:** 170 minutos  
**Sección:** Todas

---

*No escribir en estos recuadros*

20,0	=	1.          6,0	2.          7,0	3.          7,0
------	---	---	---	---

**Instrucciones:**

- a) No se permite el uso de: Internet, Laptop, Tablet, Celular, USB.
- b) No se permite el uso de material de consulta. Puede utilizar su calculadora, pero está prohibido prestarse material. Usted puede emplear las páginas de la izquierda como borrador, todo lo que se escriba en ellas no será calificado. Por ningún motivo debe arrancar estas hojas.

Número de Computadora

- c) Anote en el recuadro el numero de la PC.

**Este dato es de vital importancia, si usted no lo indica, no habrá manera de saber cual es su examen y por tanto no podrá ser calificado y Ud. Tendrá como nota CERO.**

- d) Grabe sus archivos en las siguientes rutas: **E:\Jueves\CC47\_PROGRAMACION\_I**  
Los proyectos se deberán de llamar:

Nombre del Proyecto	Contenido
P1	Grabe aquí el programa de la pregunta 1
P2	Grabe aquí el programa de la pregunta 2
P3	Grabe aquí el programa de la pregunta 3

Recuerde que la manera apropiada de crear y grabar su proyecto forma parte de su evaluación. **En caso que su proyecto no este grabado en la ruta antes mencionada no se calificará su examen.**

- e) Durante el examen, solo puede utilizar un Entorno de Desarrollo C++ (VS2015) y el Explorador Windows.

***Al finalizar el examen el alumno deberá entregar el texto del examen.***

### PREGUNTA 1 (6 puntos) – Estructuras de control repetitiva

La serie numérica proveniente del problema de Basilea (en honor al matemático Euler) queda definida como:

$$\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots = \frac{\pi^2}{6}$$

La suma exacta de la serie es  $\frac{\pi^2}{6}$  como se muestra en la definición anterior. Se le solicita a usted un programa en C++ y en entorno consola que calcule el valor de la sumatoria de los términos de la serie, si se tiene como datos el número de términos de la serie (n) y un valor **error** que hará referencia al mínimo error que debe existir entre la suma hallada y la suma exacta (umbral). El cálculo de la sumatoria finaliza si la diferencia entre la suma acumulada de términos y la suma exacta es menor al error (Ejemplo 2) o se suman los n términos (Ejemplo 1).

Recuerde que **n** es un número entero positivo y **error** es un número decimal entre 0 y 1 que debe tener máximo 7 decimales, por ejemplo, 0.0000001

Si **n** = 2 entonces:

$$\frac{1}{1^2} + \frac{1}{2^2} = 1.25$$

Si **n** = 10 entonces:

$$\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \frac{1}{5^2} + \frac{1}{6^2} + \frac{1}{7^2} + \frac{1}{8^2} + \frac{1}{9^2} + \frac{1}{10^2} = 1.54977$$

No olvide que el programa **deberá realizar todas las validaciones necesarias.**

#### Ejemplo 1:

Ingrese el valor de N: -6

Ingrese el valor de N: 100

Ingrese el mínimo error de umbral: 0.0000001

La suma aproximada es: 1.63488

#### Ejemplo 2:

Ingrese el valor de N: 100

Ingrese el mínimo error de umbral: 0.0000001

La suma aproximada es: 1.54977

Ítems	Puntajes
Lectura de datos de entrada	0.5
Validación	0.5
Cálculo de los términos de la serie	0.75
Uso de estructuras repetitivas y selectivas para calcular la serie	1.0
Calcular la suma de los términos de la serie según las condiciones	2.0
Impresión de la suma	0.5
Programa terminado y ejecución sin errores	0.75

## PREGUNTA 2 (7 puntos) – Estructuras de control repetitiva

Una empresa de radio difusión necesita las estadísticas de todas órdenes de publicidad de los clientes. Por cada orden de publicidad se debe ingresar los siguientes datos:

Tipo de cliente (N: natural; J: jurídico)

Tipo de orden (P: Pauta Publicitaria Grabada; V: Locución en vivo)

Horario de Emisión: Mañana, Tarde o Noche (M, T, N)

Cantidad de publicidad a emitir por Horario (Entero positivo entre 1 y 10)

Se le pide a usted u programa, en C++ y en entorno consola, que solicite de cada orden de publicidad los datos indicados. Luego de registrar todas las órdenes debe mostrar la cantidad de clientes de cada tipo que acceden, hallar para cada horario de emisión el número de órdenes así como determinar en que horario o horarios se realizaran más órdenes de publicidad por tipo de orden (no importa los turnos).

La lectura de datos finalizará cuando se ingrese la letra F en el tipo de cliente, luego de lo cual deberá imprimir el reporte solicitado.

Recuerde que el programa **deberá realizar todas las validaciones necesarias.**

### Ejemplo:

Ingrese tipo de cliente: N

Ingrese tipo de publicidad: P

Turno: T

Cantidad: 5

Ingrese tipo de cliente: N

Ingrese tipo de publicidad: V

Turno: M

Cantidad: 4

Ingrese tipo de cliente: N

Ingrese tipo de publicidad: P

Turno: T

Cantidad: 7

Ingrese tipo de cliente: F (Termina de solicitar datos)

Los resultados son:

-----

Tipo de cliente Natural: 2

Tipo de cliente Jurídico: 1

Órdenes de publicidad registradas en la mañana: 1

Órdenes de publicidad registradas en la tarde: 2

Órdenes de publicidad registradas en la noche: 0

El horario donde se registran más órdenes de publicidad: tarde

La cantidad promedio por Pauta Publicitaria Grabada: 6

La cantidad promedio por Locución en vivo: 4

Ítems	Puntajes
Lectura de datos de entrada	0.5
Validación	1.0
Uso de estructuras repetitivas y selectivas correctamente	1.0
Realizar el conteo de tipos de clientes	1.0
Determinación del horario u horarios con más accesos	1.0
Determinación del promedio por tipo de orden	1.5
Impresión de resultados	0.5
Programa terminado y ejecución sin errores	0.5

### PREGUNTA 3 (7 puntos) – Estructuras de control repetitiva

Elabore un programa en C++ que solicite el ingreso de un número entero positivo menor a 15 y mayor o igual a 1 y que luego nos permita visualizar varios triángulos de asteriscos al centro de la pantalla con las siguientes formas, tal como se aprecia en el ejemplo adjunto.

Recuerde que el programa **deberá realizar todas las validaciones necesarias.**

Ejemplo:

Ingrese un número: 20

Error

Ingrese un número: 4

```
      *
    ***
  *****
*****
      *
    ***
  *****
*****
      *
    ***
  *****
*****
```

Columnas														
.....	34	35	36	37	38	39	40	41	42	43	44	45	46	.....
							*							
					*	*	*							
			*	*	*	*	*							
	*	*	*	*	*	*	*							
							*							
							*	*	*					
							*	*	*	*	*			
							*	*	*	*	*	*	*	
							*							
						*	*	*						
					*	*	*	*	*					
				*	*	*	*	*	*	*				

**Nota:** En la solución del problema no debe hacer uso de la instrucción:

**Console::SetCursorPosition**

Ítems	Puntajes
Lectura de datos de entrada	0.5
Validación	0.5
Uso de estructuras repetitivas correctamente	2.0
Ubicación del gráfico en el centro de la pantalla	1.0
Impresión de los triángulos	2.5
Programa terminado y ejecución sin errores	0.5