



1. Elabore un programa que imprima en pantalla tres caracteres estáticos en sus respectivas posiciones(x,y), diferentes y distribuidos en la parte superior de la consola (ver imagen 1). Al presionar una de las teclas vocales, los tres caracteres descenden y rebotan verticalmente de manera indefinida. Luego, cada vez que se presiona la tecla "X" se elimina aleatoriamente un carácter de la pantalla. Cuando se eliminan los tres caracteres aparece un mensaje al centro de la consola que dice "La animación ha finalizado" y el programa termina. El estudiante define las dimensiones de la ventana de consola.

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

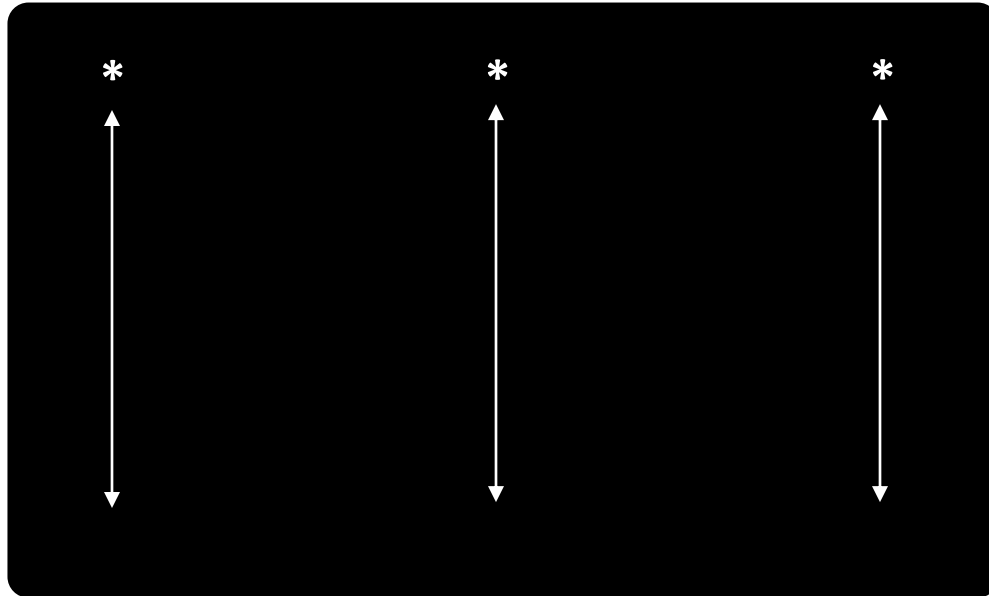


Imagen 1

2. Elabore un programa que imprima en pantalla tres caracteres estáticos en sus respectivas posiciones(x,y), diferentes (ver imagen 1). Al presionar la barra espaciadora (ascii 32), el primer carácter avanza horizontalmente hasta llegar al límite derecho. Cuando este carácter llegue al límite derecho, el segundo carácter empieza a moverse hasta llegar al límite derecho, por último, luego que el segundo carácter haya llegado al límite, el tercer carácter se mueve horizontalmente hasta llegar al límite derecho. Una vez que los tres caracteres hayan llegado al límite debe aparecer un mensaje centrado en la pantalla. El mensaje es: "La animación ha finalizado" y el programa se cierra. El estudiante define las dimensiones de la ventana de consola.

Nota. El programa debe tener, obligatoriamente, funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

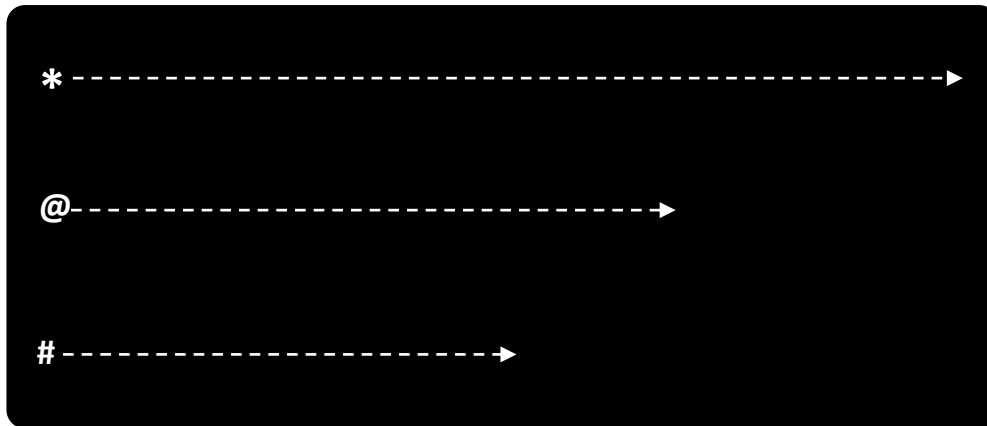


Imagen 1

3. Elabore un programa que, haciendo uso de las teclas direccionales, pueda mover un carácter por toda la pantalla sin salir de los límites. El tamaño de la ventana de consola debe ser 80x40. Cada vez que el carácter pasa por una posición(x,y) y la suma de "x"+"y" es igual a un número primo, en ese instante otro caracter debe partir de esa posición y se moverá automáticamente por toda la pantalla y si llega a uno de los de la ventana, desaparece. Si ambos caracteres colisionan en algún momento, debe aparecer un mensaje al centro de la pantalla que dice "COLISION" y luego termina el programa.

Nota. El programa debe tener, obligatoriamente, funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Código ASCII de las teclas:

Arriba:82

Abajo:80

Derecha:77

Izquierda:75

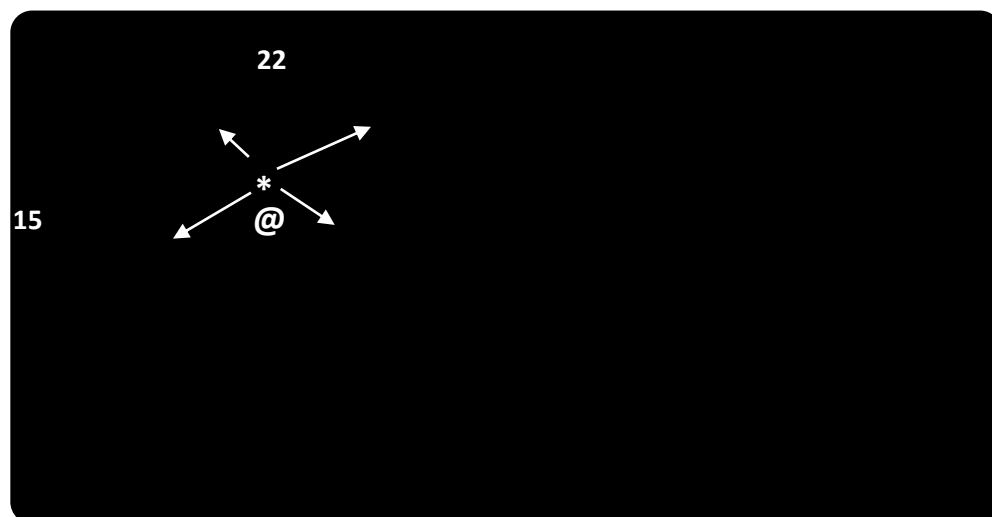


Imagen 1



Introducción a los Algoritmos
Hoja de Ejercicios
Semana 6_C

Temas: Programación
en C++ estructuras
repetitivas

4. Elabore un programa que simule un ratón (carácter 178) que debe comer pedazos de queso que están en posiciones al azar en la pantalla (ANCHO = 80 y ALTO = 40), el ratón se mueve a través de las teclas direccionales →(DERECHA 77), ←(IZQUIERDA, 75), ↓(ABAJO 80) y ↑(ARRIBA 72). Existirán 5 pedazos de queso distribuidos aleatoriamente en la pantalla. El ratón se deberá dibujar empleando caracteres ASCII: 220. Algunos pedazos de queso estarán envenenados (azar), el juego consiste en hacer que el ratón coma los pedazos de queso uno a uno, si el queso no está envenenado, se acumularan 5 puntos (el puntaje deberá ser mostrado en la pantalla, en la parte superior de esta), el juego durará mientras el ratón no coma un queso envenenado, o se coma todos los quesos buenos (el número de quesos buenos se mostrará en la pantalla, en la parte superior de esta, e irá disminuyendo conforme el ratón vaya comiendo los quesos buenos).

Nota. El programa debe tener, obligatoriamente, funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

5. Escriba un programa que muestre tres caracteres tal como se muestra en la imagen 1. Estos tres caracteres deben moverse de izquierda a derecha hasta la meta, usted puede definir las ubicaciones de los caracteres en la pantalla a su criterio al igual que la meta, teniendo en cuenta que la distancia debe ser semejante a la mostrada en la imagen. Al presionar la letra "S" los tres caracteres deben moverse a la vez y en forma automática hasta la meta. Cada carácter tiene una velocidad aleatoria las cuales serán valores enteros entre 3 y 6. El juego se detiene cuando todos los caracteres han llegado a la meta. Una vez que el juego termine debe indicar al ganador en función al orden de llegada:

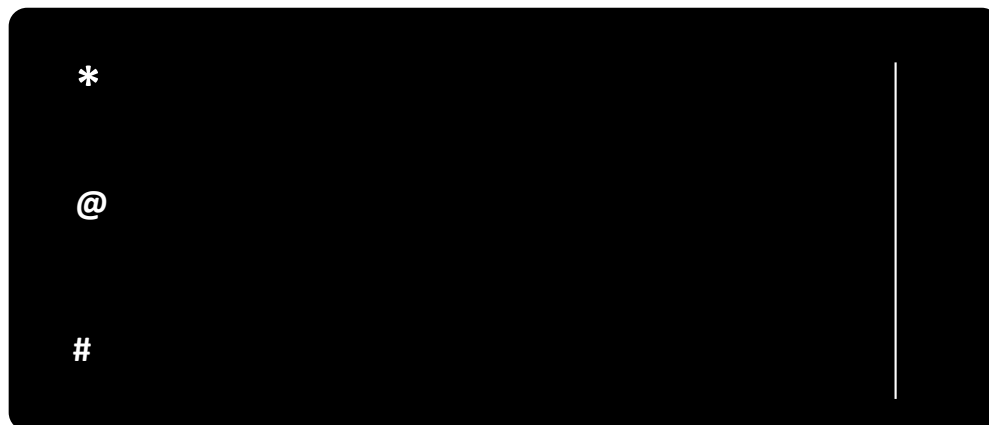


Imagen 1



6. De un conocido juego de arcade, se pide representar el movimiento de un tanque que es representado por un carácter y se mueve con las teclas direccionales (derecha, izquierda, arriba y abajo). En la pantalla tendrá 2 enemigos que estarán representados por otros dos caracteres y deben estar posicionados en forma aleatoria. El tanque lanzará una bala “*”, dirigido a los enemigos, si la bala impacta a uno de los caracteres, este desaparece de la pantalla. Si elimina todos los enemigos debe aparecer el mensaje “VICTORIA” al centro de la pantalla. El estudiante define las dimensiones de la ventana de consola.

Nota. El programa debe tener, obligatoriamente, funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo

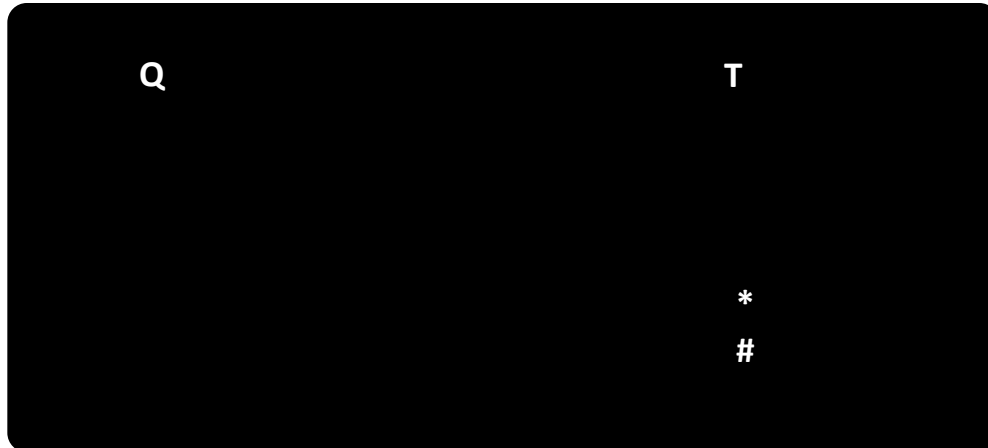


Imagen 1

7. De un conocido juego de arcade, se pide representar el movimiento de un hombre bomba con un carácter elegido por el alumno. Este personaje se mueve con las teclas direccionales (derecha, izquierda, arriba y abajo) y tendrá por misión eliminar a 2 enemigos que son otros caracteres elegidos por el alumno. El hombre bomba dejará una bomba “*”, cerca de los enemigos y después deberá detonar las bombas con la barra espaciadora = 32. Al hacerlo, el enemigo más cercano a la bomba debe desaparecer de la pantalla. Si el personaje elimina los enemigos debe aparecer al centro de la ventana el mensaje “MISIÓN CUMPLIDA”

Nota. El programa debe tener, obligatoriamente, funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo

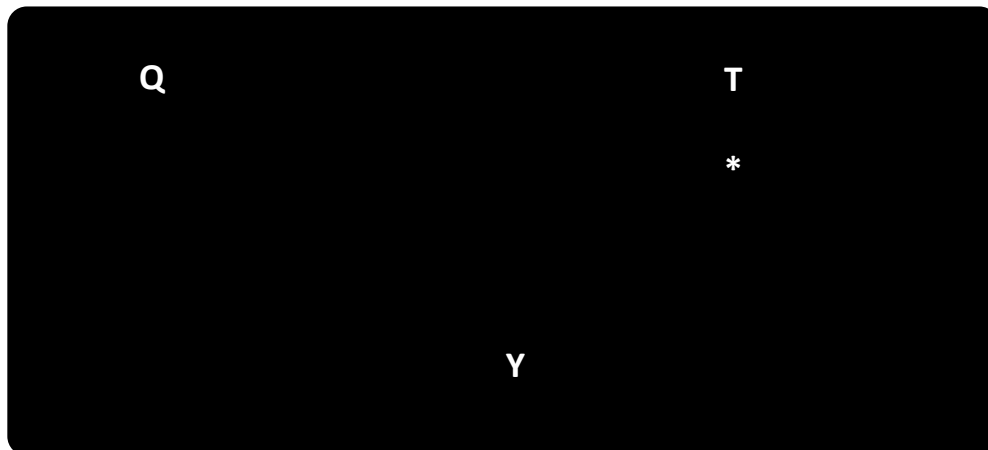


Imagen 1



8. Una compañía de videojuegos retro está desarrollando un juego en formato de consola en C++ que incluya animaciones básicas con tres móviles representados por caracteres ASCII. Estos móviles se desplazan en un panel de 80 columnas y 25 filas. Se le pide que desarrolle un programa que simule el movimiento de los móviles en el panel. Los móviles son: Nave espacial: Representada por el caracter '^'. Asteroide: Representado por el caracter 'O' y Estrella: Representada por el caracter '*'

Cada móvil tiene su propio movimiento que será un algoritmo propuesto por el estudiante. La nave espacial controla con las teclas direccionales o algún caracter (por ejemplo, W para arriba, A para izquierda, S para abajo y D para derecha).

Se requiere lo siguiente:

- Inicializar el panel con los móviles en posiciones aleatorias.
- Mover automáticamente el asteroide y la estrella siguiendo un algoritmo propuesto por usted.
- Permitir que el usuario controle el movimiento de la nave espacial utilizando teclas direccionales o caracteres específicos.

Nota. El programa debe tener, obligatoriamente, funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

9. Elabore un programa que al iniciar muestre la pantalla vacía. Luego, cada vez que se presiona la tecla "P" debe aparecer un carácter en una posición par x,y aleatoria. Es decir, coordenada "x" par e "y" par. Si se presiona la tecla "I" aparecer un carácter en una posición impar x,y aleatoria. Es decir, coordenada "x" impar e "y" impar. El carácter que aparece debe quedarse estático y el número máximo de caracteres que aparecen es tres. Cuando los tres caracteres hayan aparecido en la pantalla y se presione la barra espaciadora, en ese momento los caracteres empiezan a moverse en direcciones opuestas unas con otras y rebotan al llegar a los bordes.

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

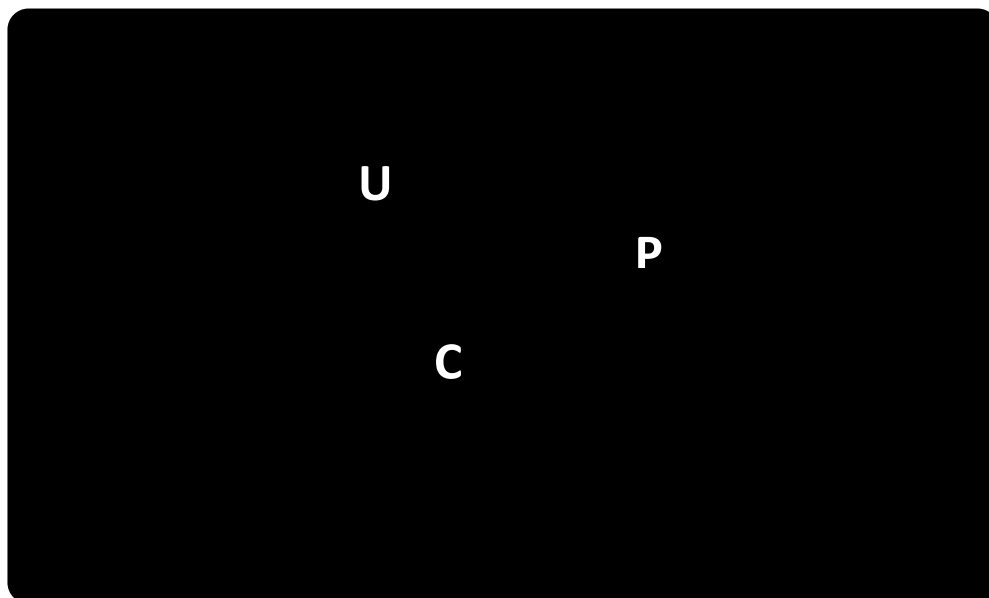




Imagen 1

10. Elabore un programa que al iniciar establezca aleatoriamente las posiciones de 3 caracteres que representan unas minas ocultas. Es decir, cada mina es un carácter y está ubicada en una posición x,y aleatoria pero no es visible en pantalla porque está oculta. Luego, un carácter que representa a un personaje, empieza a moverse con las teclas direccionales. Si este personaje pasa por la posición de una mina, debe retornar a su posición inicial. Si el personaje llega a pasar por las tres minas el programa muestra un mensaje de "FIN" al centro de la pantalla.

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

11. Elabore un programa que solicite un número "n" en el intervalo de 2 a 6 y que dibuje la figura como se ve en la Imagen 1

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo:

Ingrese un número: 4

	1	2	3	4	5	6	7	8	9	10	
1	A						A				
2	B	A				A	B				
3	C	B	A		A	B	C				
4	D	C	B	A	B	C	D				
5		D	C	B	C	D					
6			D	C	D						
7				D							
8											
9											

Ingrese un número: 5

		1	2	3	4	5	6	7	8	9	10	
	1	A								A		
	2	B	A						A	B		
	3	C	B	A				A	B	C		
	4	D	C	B	A		A	B	C	D		
	5	E	D	C	B	A	B	C	D	E		
	6		E	D	C	B	C	D	E			
	7			E	D	C	D	E				
	8				E	D	E					
	9					E						

Imagen 1

12. Elabore un programa que solicite un número "n" en el intervalo de 4 a 7 y que dibuje la figura de dimensión "n" como se ve en la Imagen 1

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ingrese un número: 5



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	18	19	20	21	22	23	24	25	26
1	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
2		5			5		5			5		5			5		5			5		5			5
3			5		5			5		5			5		5			5		5			5		5
4				5	5				5	5				5	5				5	5				5	5
5					5					5					5					5					5

Ingrese un número: 4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
2		4		4		4		4		4		4		4		4
3			4	4			4	4			4	4			4	4
4				4				4				4				4

Imagen 1

13. Elabore un programa que solicite un número “n” en el intervalo de 2 a 6 y que dibuje las siguientes “n” figuras, tal como se muestra en la Imagen 1

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo:

Ingrese un número: 4

	1	2	3	4	5	6	7	8	9	
1			a							
2		a	e	i						
3	a	e	i	o	u					
4		a	e	i						
5			a							
6			a							
7		a	e	i						
8	a	e	i	o	u					
9		a	e	i						
10			a							
11			a							
12		a	e	i						
13	a	e	i	o	u					
14		a	e	i						
15			a							
16			a							
17		a	e	i						
18	a	e	i	o	u					
19		a	e	i						

Ingrese un número: 5

	1	2	3	4	5	6	7	8	9	
1			a							
2		a	e	i						
3	a	e	i	o	u					
4		a	e	i						
5			a							
6			a							
7		a	e	i						
8	a	e	i	o	u					
9		a	e	i						
10			a							
11			a							
12		a	e	i						
13	a	e	i	o	u					
14		a	e	i						
15			a							
16			a							
17		a	e	i						
18	a	e	i	o	u					
19		a	e	i						



20			a							
21										
22										
23										
24										
25										

20			a							
21			a							
22		a	e	i						
23	a	e	i	o	u					
24		a	e	i						
25			a							

Imagen 1

14. Elabore un programa que solicite un número “n” en el intervalo de 2 a 6 y que dibuje la siguiente figura, tal como se ve en la Imagen 1.

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo:

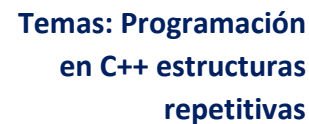
Ingrese un número: 4

	1	2	3	4	5	6	7	8	9	10	11
1							4				
2						4	4				
3					4		4				
4				4	4	4	4				
5						3					
6					3	3					
7				3		3					
8			3	3	3	3					
9					2						
10				2	2						
11			2		2						
12		2	2	2	2						
13				1							
14			1	1							
15		1		1							
16	1	1	1	1							

Ingrese un número: 5

	1	2	3	4	5	6	7	8	
1								5	
2							5	5	
3						5		5	
4					5	5	5	5	
5							4		
6						4	4		
7					4		4		
8				4	4	4	4		
9						3			
10					3	3			
11				3		3			
12			3	3	3	3			
13					2				
14				2	2				
15			2		2				
16		2	2	2	2				
17				1					
18			1	1					
19		1		1					
20	1	1	1	1					

Imagen 1



- Nota.** El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ingrese el valor de n: 3

	1	2	3	4	5	6	7
1	A						A
2	A	A				A	A
3	A		A		A		A
4	A			A			A
5	A		A		A		A
6	A	A				A	A
7	A						A
8	E						E
9	E	E				E	E
10	E		E		E		E
11	E			E			E
12	E		E		E		E
13	E	E				E	A
14	E						E
15	I						I
16	I	I				I	I
17	I		I		I		I
18	I			I			I
19	I		I		I		I
20	I	I				I	I
21	I						I

Imagen 1



16. Escribir un programa en el que dado la dimensión del lado m (m mayor a 3 y menor a 10), nos muestre m figuras iguales, tal como se ve en la Imagen 1

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ingrese m: 4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	o	o	o	o	o	o	o		o	o	o	o	o	o	o		o	o	o	o	o	o	o		o	o	o	o	o	o	o
2	o	o				o	o		o	o				o	o		o	o				o	o		o	o				o	o
3	o		o		o		o		o		o		o		o		o		o		o		o		o		o		o		o
4	o			o			o		o			o			o		o			o			o		o			o			o
5	o		o		o		o		o		o		o		o		o		o		o		o		o		o		o		o
6	o	o				o	o		o	o				o	o		o	o				o	o		o	o				o	o
7	o	o	o	o	o	o	o		o	o	o	o	o	o	o		o	o	o	o	o	o	o		o	o	o	o	o	o	o

Ingrese m: 3

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	o	o	o	o	o	o	o		o	o	o	o	o	o	o		o	o	o	o	o	o	o
2	o	o				o	o		o	o				o	o		o	o				o	o
3	o		o		o		o		o		o		o		o		o		o		o		o
4	o			o			o		o			o			o		o			o			o
5	o		o		o		o		o		o		o		o		o		o		o		o
6	o	o				o	o		o	o				o	o		o	o				o	o
7	o	o	o	o	o	o	o		o	o	o	o	o	o	o		o	o	o	o	o	o	o

Imagen 1



Introducción a los Algoritmos
Hoja de Ejercicios
Semana 6_C

Temas: Programación
en C++ estructuras
repetitivas

17. Escribir un programa en el que dado la dimensión del lado m (m mayor a 3 y menor a 10), nos muestre m figuras iguales, tal como se ve en la Imagen 1.

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo

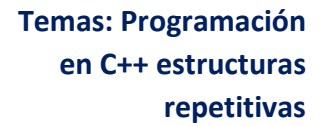
Ingrese m: 4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	o				o				o				o										
2	o	o			o	o			o	o			o	o									
3	o		o		o		o		o		o		o		o								
4	o		1	o	o		2	o	o		3	o	o		4	o							
5	o		o		o		o		o		o		o		o								
6	o	o			o	o			o	o			o	o									
7	o				o				o				o										

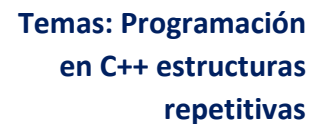
Ingrese m: 6

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	
1	0						0						0						0						0						0						
2	0	0					0	0					0	0					0	0					0	0					0	0					
3	0		0				0		0				0		0				0		0				0		0				0		0				
4	0			0			0			0			0			0			0			0			0			0			0			0			
5	0				0		0				0		0				0		0				0		0				0		0				0		
6	0				1	0	0				2	0	0				3	0	0				4	0	0				5	0	0				6	0	
7	0				0		0				0		0				0		0				0		0				0		0				0		
8	0			0			0			0			0			0			0			0			0			0			0			0			
9	0		0				0		0				0		0				0		0				0		0				0		0				
10	0	0					0	0					0	0					0	0					0	0					0	0					
11	0						0						0						0						0						0						

Imagen 1



- ### Imagen 1



Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

	1	2	3	4	5	6	7	8	9	10	11	12	
1	*	*	*	*									
2	*	*	*	*									
3	*	1	*	*									
4	*	*	*	*									
5													
6	*	*	*	*									
7	*	*	*	*									
8	*	2	*	*									
9	*	*	*	*									
10					*	*	*	*					
11					*	*	*	*					
12					*	2	*	*					
13					*	*	*	*					
14	*	*	*	*									
15	*	*	*	*									
16	*	3	*	*									
17	*	*	*	*									
18					*	*	*	*					
19					*	*	*	*					
20					*	3	*	*					
21					*	*	*	*					
22									*	*	*	*	
23									*	*	*	*	
24									*	3	*	*	
25									*	*	*	*	

[illegible]

Imagen 1



Introducción a los Algoritmos
Hoja de Ejercicios
Semana 6_C

Temas: Programación
en C++ estructuras
repetitivas

20. Elabore un programa que solicite un número “n” en el intervalo de 3 a 6 y que dibuje las siguientes “n” figuras, tal como se ve en la Imagen 1

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ejemplo

Ingrese n: 3

	1	2	3	4	5	6	7	8	9	10	11	12
1	*	*	*	*	*	*	*					
2	*			*			*					
3	*		*		*		*					
4	*	*				*	*					
5	*			1			*					
6	*	*				*	*					
7	*		*		*		*					
8	*			*			*					
9	*	*	*	*	*	*	*					
10			*	*	*	*	*	*	*			
11			*			*			*			
12			*		*		*		*			
13			*	*				*	*			
14			*			2			*			
15			*	*				*	*			
16			*		*		*		*			
17			*			*			*			
18			*	*	*	*	*	*	*			
19					*	*	*	*	*	*	*	*
20					*			*			*	
21					*		*		*		*	
22					*	*				*	*	
23					*			3			*	
24					*	*				*	*	
25					*		*		*		*	
26					*			*			*	
27					*	*	*	*	*	*	*	*

Ingrese n: 4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	*	*	*	*	*	*	*							
2	*			*			*							
3	*		*		*		*							
4	*	*				*	*							
5	*			1			*							
6	*	*				*	*							
7	*		*		*		*							
8	*			*			*							
9	*	*	*	*	*	*	*							
10			*	*	*	*	*	*	*					
11			*			*			*					
12			*		*		*		*					
13			*	*				*	*					
14			*			2			*					
15			*	*				*	*					
16			*		*		*		*					
17			*			*			*					
18			*	*	*	*	*	*	*	*				
19					*	*	*	*	*	*	*	*	*	*
20					*			*			*			
21					*		*		*		*			
22					*	*				*	*			
23					*			3			*			
24					*	*				*	*			
25					*		*		*		*			
26					*			*			*			
27					*	*	*	*	*	*	*	*	*	*
28							*	*	*	*	*	*	*	*
29							*			*			*	*
30							*	*		*		*	*	*
31							*	*			*	*	*	*
32							*			4			*	*
33							*	*			*	*	*	*
34							*		*		*		*	*
35							*		*		*		*	*
36							*	*	*	*	*	*	*	*

Imagen 1



Introducción a los Algoritmos
Hoja de Ejercicios
Semana 6_C

Temas: Programación
en C++ estructuras
repetitivas

21. Elabore un programa en Lenguaje C++ que solicite el ingreso de un número n ($3 \leq n \leq 9$) y realice el siguiente gráfico, tal como se ve en la Imagen 1

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ingrese n: 3

	1	2	3	4	5	6	7	8	9	10		
1	2				2				2			
2		2			2			2				
3			2		2		2					
4				2	2	2						
5				2	2	2						
6	3				3				3			
7		3			3			3				
8			3		3		3					
9				3	3	3						
10				3	3	3						
11	4				4				4			
12		4			4			4				
13			4		4		4					
14				4	4	4						
15				4	4	4						
16												
17												
18												
19												
20												

Ingrese n: 4

	1	2	3	4	5	6	7	8	9
1	2				2				2
2		2			2			2	
3			2		2		2		
4				2	2	2			
5				2	2	2			
6	3				3				3
7		3			3			3	
8			3		3		3		
9				3	3	3			
10				3	3	3			
11	4				4				4
12		4			4			4	
13			4		4		4		
				4	4	4			
				4	4	4			
	5				5				5
		5			5			5	
			5		5		5		
				5	5	5			
				5	5	5			

Imagen 1



22. Elabore un programa en Lenguaje C++ que solicite el ingreso de un número n ($4 \leq n \leq 9$) y realice el siguiente gráfico, tal como se ve en la Imagen 1.

Nota. El programa debe incluir el uso obligatorio de funciones creadas por el estudiante y todas las validaciones necesarias. Debe estar bien estructurado, siguiendo buenas prácticas de programación en C++.

Ingrese n: 5

	1	2	3	4	5	6	7	8	9	10		
1	*											
2	*	*										
3	*	*										
4	*		*									
5	*	*	*									
6	*		*									
7	*			*								
8	*		*	*								
9	*	*		*								
10	*			*								
11	*				*							
12	*			*	*							
13	*		*		*							
14	*	*			*							
15	*				*							
16												
17												
18												
19												
20												

Ingrese n: 6

	1	2	3	4	5	6	7	8	9
1	*								
2	*	*							
3	*	*							
4	*		*						
5	*	*	*						
6	*		*						
7	*			*					
8	*		*	*					
9	*	*		*					
10	*			*					
11	*				*				
12	*			*	*				
13	*		*		*				
14	*	*			*				
15	*				*				
16	*					*			
17	*				*	*			
18	*			*		*			
19	*		*			*			
20	*	*				*			
21	*					*			

Imagen 1