

TALLER REFUERZO 2 – PSEUDOCÓDIGO

Tema 1: Algoritmos básicos de entrada y salida

Objetivos

- Identificar la estructura básica de un algoritmo.
- Familiarizar al estudiante con las fases del desarrollo de un algoritmo.
- Familiarizar al estudiante con la identificación de la sintaxis de un programa en C++.
- Reconocer la importancia de los tipos de datos en un lenguaje de programación.
- Utilizar las instrucciones aritméticas para realizar cálculos básicos en C++.
- Comprender el uso y la sintaxis de las operaciones de lectura y escritura mediante la función **cin()** y la función **cout()** - **cerr()**.

Metodología para resolver problemas

1. Analizar el problema
2. Diseñar el algoritmo y escribirlo en pseudocódigo
3. Realizar una prueba de escritorio para el algoritmo
4. Codificar el algoritmo en el lenguaje de programación
5. Probar el algoritmo implementado

Ejercicio resuelto

Problema 1

Desarrollar un programa que solicita el radio de un círculo y calcula el diámetro y el área.

Solución

Paso 1: Analizar el problema

Entradas	Salidas
<div>Digite el radio: 2.3</div>	<div>Diámetro: 4.6 Área: 16.6106</div>

Paso 2: Diseñar el algoritmo y escribirlo en pseudocódigo

Pseudocódigo

Inicio

radio : real

diametro, area : real

preguntar (radio)

diametro = 2*radio

area = 3.14*radio*radio

mostrar (diametro)

mostrar (area)

Fin

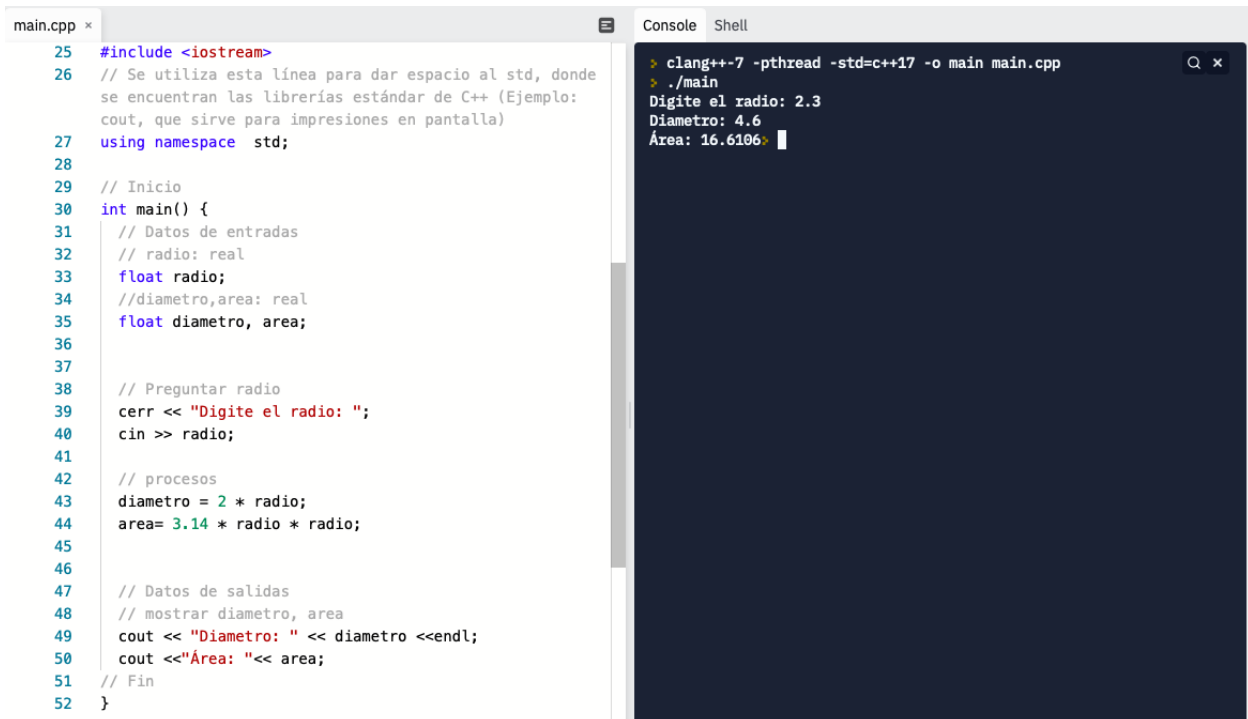
Paso 3: Realizar una prueba de escritorio para el algoritmo

radio	diametro	area
2.3	4.6	16.6106
4	8	50.24
3	6	28.26

Paso 4: Codificar el algoritmo en el lenguaje de programación C++

```
main.cpp x
25 #include <iostream>
26 // Se utiliza esta línea para dar espacio al std, donde
   // se encuentran las librerías estándar de C++ (Ejemplo:
   // cout, que sirve para impresiones en pantalla)
27 using namespace std;
28
29 // Inicio
30 int main() {
31     // Datos de entradas
32     // radio: real
33     float radio;
34     //diametro,area: real
35     float diametro, area;
36
37
38     // Preguntar radio
39     cerr << "Digite el radio: ";
40     cin >> radio;
41
42     // procesos
43     diametro = 2 * radio;
44     area= 3.14 * radio * radio;
45
46
47     // Datos de salidas
48     // mostrar diametro, area
49     cout << "Diametro: " << diametro <<endl;
50     cout <<"Área: "<< area;
51 // Fin
52 }
```

Paso 5: Probar el algoritmo implementado



The image shows a code editor with a file named `main.cpp` and a terminal window. The code in `main.cpp` is a C++ program that calculates the diameter and area of a circle based on a user-provided radius. The terminal shows the compilation and execution of the program, with the user inputting a radius of 2.3 and the program outputting a diameter of 4.6 and an area of 16.6106.

```
25 #include <iostream>
26 // Se utiliza esta línea para dar espacio al std, donde
   // se encuentran las librerías estándar de C++ (Ejemplo:
   // cout, que sirve para impresiones en pantalla)
27 using namespace std;
28
29 // Inicio
30 int main() {
31     // Datos de entradas
32     // radio: real
33     float radio;
34     //diámetro, área: real
35     float diametro, area;
36
37
38     // Preguntar radio
39     cerr << "Digite el radio: ";
40     cin >> radio;
41
42     // procesos
43     diametro = 2 * radio;
44     area = 3.14 * radio * radio;
45
46
47     // Datos de salidas
48     // mostrar diametro, area
49     cout << "Diametro: " << diametro << endl;
50     cout << "Área: " << area;
51 // Fin
52 }
```

```
> clang++-7 -pthread -std=c++17 -o main main.cpp
> ./main
Digite el radio: 2.3
Diametro: 4.6
Área: 16.6106
```

Replit: <https://replit.com/@FDP1/CalcularDyA#main.cpp>

Ejercicios propuestos (INGInious)

Desarrolle los siguientes ejercicios de INGInious:

- Ejercicio 04 – Programa Minicalculadora
<http://ingin.ddns.net/course/FDP-2022-2/Ejercicio04>
- Ejercicio 05 – Programa Netflix
<http://ingin.ddns.net/course/FDP-2022-2/Ejercicio05>

Para su entrega es necesario documentar el código fuente con los 5 pasos descritos en la metodología y tener en cuenta los casos de prueba (entradas y salidas).

Tema 2: Estructuras de decisión

Objetivos

- Identificar los operadores relacionales y lógicos.
- Construir y aplicar expresiones condicionales.
- Familiarizar al estudiante con las distintas estructuras condicionales en C++.

Metodología para resolver problemas

1. Analizar el problema
2. Diseñar el algoritmo y escribirlo en pseudocódigo
3. Realizar una prueba de escritorio para el algoritmo
4. Codificar el algoritmo en el lenguaje de programación
5. Probar el algoritmo implementado

Ejercicios resueltos (guía asistida)

Problema 1

Desarrollar un programa que solicite un número entero y determine si es positivo. En el caso que sea negativo o cero no debe mostrar ningún mensaje.

Solución

Paso 1: Analizar el problema

Entradas	Salidas
<div>Digite un número: 24</div>	<div>El número es positivo</div>

Paso 2: Diseñar el algoritmo y escribirlo en pseudocódigo

Pseudocódigo

Inicio

numero : entero

preguntar (numero)

si (numero>0)

mostrar ("El número es positivo")

Fin

Paso 3: Realizar una prueba de escritorio para el algoritmo

numero	salida
24	El número es positivo
-140	

Paso 4: Codificar el algoritmo en el lenguaje de programación C++

```
20 #include <iostream>
21 using namespace std;
22
23 // Inicio
24 int main() {
25     // Datos de entradas
26     // numero: entero
27     int numero;
28
29     // Preguntar numero
30     cerr << "Ingrese el numero: ";
31     cin >> numero;
32
33     // Proceso y Salida
34     if (numero>0){
35         cout << "El numero es positivo";
36     }
37     // Fin
38 }
```

Paso 5: Probar el algoritmo implementado

```
20 #include <iostream>
21 using namespace std;
22
23 // Inicio
24 int main() {
25     // Datos de entradas
26     // numero: entero
27     int numero;
28
29     // Preguntar numero
30     cerr << "Ingrese el numero: ";
31     cin >> numero;
32
33     // Proceso y Salida
34     if (numero>0){
35         cout << "El numero es positivo";
36     }
37     // Fin
38 }
```

```
> clang++-7 -pthread -std=c++17 -o main main.cpp
> ./main
Ingrese el numero: 24
El numero es positivo>
```

Replit: <https://replit.com/@FDP1/Positivo#main.cpp>

Ejercicios propuestos (INGInious)

Desarrolle los siguientes ejercicios de INGInious:

- Ejercicio 08 – Calcular área de un cuadrado o de un círculo
<http://ingin.ddns.net/course/FDP-2022-2/Ejercicio08>
- Ejercicio 09 – Librería
<http://ingin.ddns.net/course/FDP-2022-2/Ejercicio09>

Para su entrega es necesario documentar el código fuente con los 5 pasos descritos en la metodología y tener en cuenta los casos de prueba (entradas y salidas).

Tema 3: Funciones

Objetivos

- Identificar los operadores relacionales y lógicos.
- Familiarizar al estudiante con las funciones en C++.
- Comprender el uso y la sintaxis de una función.

Metodología para resolver problemas

1. Analizar el problema
2. Diseñar el algoritmo y escribirlo en pseudocódigo
3. Realizar una prueba de escritorio para el algoritmo
4. Codificar el algoritmo en el lenguaje de programación
5. Probar el algoritmo implementado

Ejercicios resueltos (guía asistida)

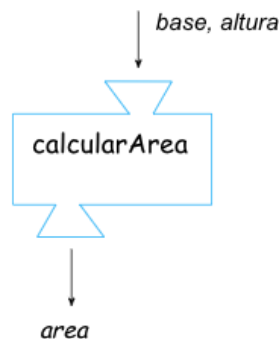
Problema 1

Desarrollar un programa que permite calcular el área de un triángulo dados su base y su altura.

Solución

Paso 1: Analizar el problema

Entradas	Salidas
Base, altura	area



Paso 2: Diseñar el algoritmo y escribirlo en pseudocódigo

Pseudocódigo

Inicio entero calcularArea(base, altura)	Inicio main()
area : real	base, altura : real
area = (base* altura) / 2	preguntar(base)
retornar(area)	preguntar(altura)
Fin	mostrar(calcularArea(base, altura))
	Fin

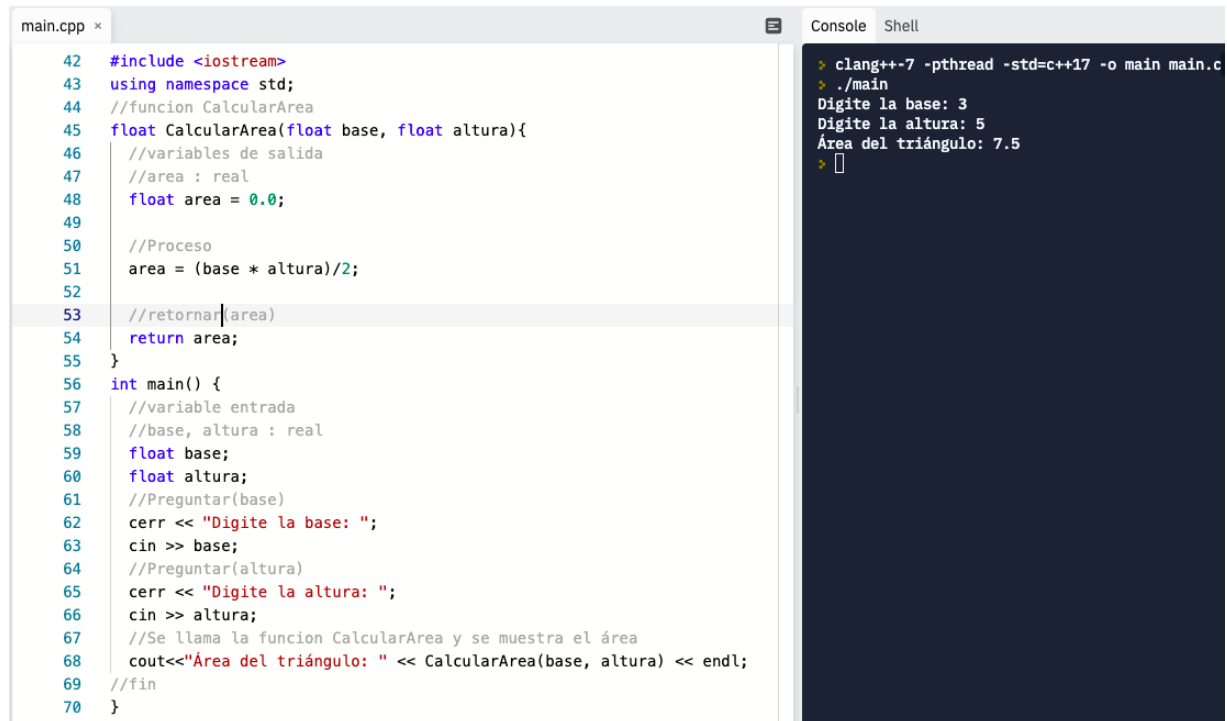
Paso 3: Realizar una prueba de escritorio para el algoritmo

base	altura	area
3	5	7.5
1	4	2

Paso 4: Codificar el algoritmo en el lenguaje de programación C++

```
main.cpp x
42  #include <iostream>
43  using namespace std;
44  //funcion CalcularArea
45  float CalcularArea(float base, float altura){
46      //variables de salida
47      //area : real
48      float area = 0.0;
49
50      //Proceso
51      area = (base * altura)/2;
52
53      //retornar(area)
54      return area;
55  }
56  int main() {
57      //variable entrada
58      //base, altura : real
59      float base;
60      float altura;
61      //Preguntar(base)
62      cerr << "Digite la base: ";
63      cin >> base;
64      //Preguntar(altura)
65      cerr << "Digite la altura: ";
66      cin >> altura;
67      //Se llama la funcion CalcularArea y se muestra el área
68      cout<<"Área del triángulo: " << CalcularArea(base, altura) << endl;
69      //fin
70  }
```


Paso 5: Probar el algoritmo implementado



The screenshot shows a C++ IDE with two panels. The left panel displays the source code for `main.cpp`, and the right panel shows the console output.

```
42 #include <iostream>
43 using namespace std;
44 //funcion CalcularArea
45 float CalcularArea(float base, float altura){
46     //variables de salida
47     //area : real
48     float area = 0.0;
49
50     //Proceso
51     area = (base * altura)/2;
52
53     //retorna: (area)
54     return area;
55 }
56 int main() {
57     //variable entrada
58     //base, altura : real
59     float base;
60     float altura;
61     //Preguntar(base)
62     cerr << "Digite la base: ";
63     cin >> base;
64     //Preguntar(altura)
65     cerr << "Digite la altura: ";
66     cin >> altura;
67     //Se llama la funcion CalcularArea y se muestra el área
68     cout<<"Área del triángulo: " << CalcularArea(base, altura) << endl;
69     //fin
70 }
```

The console output shows the execution of the program:

```
> clang++-7 -pthread -std=c++17 -o main main.c
> ./main
Digite la base: 3
Digite la altura: 5
Área del triángulo: 7.5
> 
```

Ejercicios propuestos (INGInious)

Desarrolle los siguientes ejercicios de INGInious:

- Ejercicio 13 – Área y perímetro de un círculo
<http://ingin.ddns.net/course/FDP-2022-2/Ejercicio13>
- Ejercicio 14 – Mayor valor entre x, y, z
<http://ingin.ddns.net/course/FDP-2022-2/Ejercicio14>

Para su entrega es necesario documentar el código fuente con los 5 pasos descritos en la metodología y tener en cuenta los casos de prueba (entradas y salidas).