




Manejo de Datos




Manejo de Datos

C++

Existen dos formas de manejar la entrada y la salida en C++:

- cin - cout
- scanf printf



Manejo de Datos

C++

Manejo de input en C++

La función que nos provee C++ para leer la entrada es:


```
istream & operator >>
```

Este operador trabaja sobre un stream de entrada. El stream asociado a la entrada estándar es `std::cin`.

La ventaja de este operador es que lee el stream hasta encontrar el formato indicado por el segundo operando, por lo que usando el mismo operador podemos leer números, strings, caracteres, etc.

```
char c ;  
  
string s;  
  
int d;  
  
std::cin >> s >> c >> d;
```

En el ejemplo se lee primero un **string**, luego un **carácter** y por último un **entero**.



Manejo de Datos

C++


Manejo de output en C++

La función que nos provee C++ para imprimir la salida es:

```
ostream & operator <<
```

Este operador trabaja sobre un stream de salida. El stream asociado a la salida estándar es `std::cout`. Trabaja de forma similar al operador de lectura pero de forma inversa.

```
char c = 's';  
string s = "Texto de prueba";  
int d = 80;  
std::cout << s << c << d;
```




Manejo de Datos

C++

Manejo de input/output en C++

Uso de scanf/printf permite un tiempo más eficiente:

Código[GNU] C++	Sample Input	Sample Output
<pre>#include<cstdio> int TC, a, b; scanf("%d", &TC); //nro de casos while(TC --){ //repite hasta 0 scanf("%d%d", &a, &b); //calcular la resp printf("%d \n", a + b); }</pre>	<pre>3 1 2 5 7 6 3</pre>	<pre>3 12 9</pre>



Manejo de Datos Java


Manejo de input en Java

Java nos provee una clase **wrapper**: `java.util.Scanner`

Esta clase provee métodos para extraer distintos tipos de datos de un stream de datos, así como para saber si existen datos en el stream. En nuestro caso usaremos `System.in` como stream que es la entrada estándar.

Para ver cómo se usa entren a: <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/Scanner.html>

```
Scanner in = new Scanner(new BufferedReader(new
    InputStreamReader
    (System.in)));
```



Manejo de Datos Java

Manejo de output en Java

El manejo de output en java se hace mediante el método `println`, `print` o `format` de la clase `PrintStream`.

El objeto sobre el cual tendremos que llamar los métodos es `System.out` el cual es un `PrintStream` que representa la salida estándar.

La semántica de las funciones es muy simple y puede verse en:

<https://docs.oracle.com/javase/7/docs/api/java/io/PrintStream.html>

```
BufferedWriter out = new BufferedWriter(new  
OutputStreamWriter(System.out));  
  
out.write("strings");  
  
out.flush() // AL FINAL DEL PROGRAMA!
```




Manejo de Datos Python

Manejo de input en Python

El manejo de input en python se hace mediante el método input() el cual lee una línea y retorna un string.

En python2.7 tenemos el mismo método pero se llama raw_input().

```
number = int(input())  
input1, input2 = map(int, input.split())  
inputArray = map(int, input.split())
```

Manejo de Datos Python

Manejo de output en Python

El manejo de output en python se hace mediante el método print() el cual puede recibir más de un parámetro e imprime los parámetros recibidos, separándolos con un espacio y agregando un salto de línea al final.

```
name = "Eric"
age = 74
print(f"Hello, {name}. You are {age}.")
'Hello, Eric. You are 74.'
```

Funciona como un wrapper del método **stdout.write**



Manejo de Datos Python

Manejo de input/output en Python


Mejora de la eficiencia usando `stdin.readline` y `stdout.write`

```
# input via readline method
n = stdin.readline()
# array input similar method
arr = [int(x) for x in stdin.readline().split()]
```

Write solo acepta Strings!

```
stdout.write(str(summation))
```

Fuente: <https://www.geeksforgeeks.org/python-input-methods-competitive-programming/>



Manejo de Datos Introducción

Datos de entrada

En todos los problemas de las competencias los datos de entrada deben ser leídos de la entrada estándar.

El formato en que vienen los datos es explicitado en el enunciado del problema.

En los problemas recientes el formato de los datos ha sido estandarizado y estructurado por lo que es raro que se encuentren con problemas donde tengan que leer toda una línea para después parsearla.



Manejo de Datos Introducción

Datos de salida

En todos los problemas de las competencias los datos de salida deben ser impresos a la salida estándar.

El formato en el que hay que imprimir los datos es explicitado en el enunciado del problema.

Prestar mucha atención a los espacios y las líneas en blanco que pide el enunciado. No realizar esto correctamente resulta en **Presentation Error** lo que perjudica innecesariamente el puntaje.