

# Guía para Timus

---

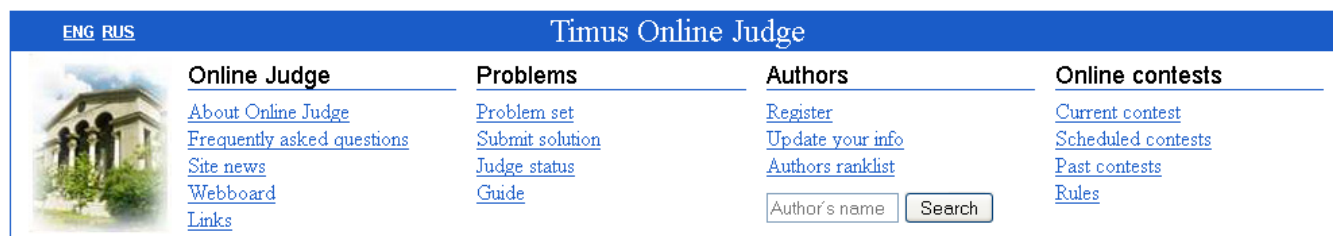
*Realizada por: Lucas Ponce de León*

<b>PASO 1: ACCESO A LA PÁGINA .....</b>	<b>2</b>
<b>PASO 2: REGISTRO .....</b>	<b>3</b>
<b>PASO 3: ACCEDER A UN PROBLEMA.....</b>	<b>4</b>
<b>PASO 4: ENVIAR SOLUCIÓN .....</b>	<b>5</b>
<b>PASO 5: VER RESULTADOS .....</b>	<b>6</b>
<b>ANEXO I: REQUISITOS DEL CÓDIGO FUENTE .....</b>	<b>7</b>
<b>ANEXO II: CLASES ANIDADAS .....</b>	<b>8</b>
<b>ANEXO III: ENTRADA Y SALIDA POR CONSOLA.....</b>	<b>9</b>
ENTRADA: .....	9
SALIDA: .....	9

## Paso 1: Acceso a la página

Para acceder al sitio del juez online Timus, debemos ingresar a la URL: <http://acm.timus.ru/>

Una vez allí, veremos una pantalla de bienvenida y con el siguiente panel:



De aquí, nos interesarán los siguientes 4 enlaces:

- **Authors → Register**
- **Problems → Problem set**
- **Problems → Submit solution**
- **Problems → Judge status**

## Paso 2: Registro

Para registrarnos, debemos acceder a: **Authors → Register**

Allí debemos completar el siguiente formulario, en el que ingresaremos nombre de usuario, clave, email, país y el código verificador. Los demás campos son opcionales.

**Register**

Name:   
Name must be given in English.

Password:

Confirm password:

Email:   
Requisites necessary for the work in the system will be sent to this email

Country:

---

Homepage:

Motto:

---

☐ Notify me about upcoming online contests

---

 Enter the verification code shown on the left:

*If you cannot read the numbers, click on the image to generate a new code.*

Una vez que no hallamos registrados, debemos ingresar al correo electrónico para buscar el mail de verificación (es posible que esté en "Correo no deseado") y en él estará el ID y la clave con la cual realizaremos los "submit" con las soluciones de los problemas.

El ID tiene un formato de 6 dígitos y 2 letras (#####AA), es importante tenerlo porque con él se hacen todos los envíos.

### Paso 3: Acceder a un problema

Una vez que ya estemos registrados, podemos ingresar a los problemas desde **Problems** → **Problem set**. Aquí se muestra un ejemplo de un problema simple de ejemplo. Se recomienda realizarlo para probar que los **submit** se realicen de forma satisfactoria:

## 1000. A+B Problem

Time limit: 1.0 second

Memory limit: 64 MB

Calculate  $a + b$

### Input

a and b

### Output

a+b

### Sample

input	output
1 5	6

### Hint

Use + operator

**Problem Author:** Pavel Atnashev

**Tags:** [problem for beginners](#) [\(hide tags for unsolved problems\)](#)

Difficulty: 20   [Printable version](#)   [Submit solution](#)   [Discussion \(98\)](#)

[All submissions \(176665\)](#)   [All accepted submissions \(90846\)](#)   [Solutions rating \(56053\)](#)

En la parte inferior, podemos ver el enlace **Submit solution**, en el cual podremos realizar el envío de la solución para este problema.

Otra forma de enviar una solución es ingresando a **Problems** → **Submit solution**. Si ingresamos de esta forma debemos indicar el ID del problema.

## Paso 4: Enviar solución

Para enviar la solución, una vez que ingresamos a **Submit solution** por cualquiera de las dos formas indicadas anteriormente, debemos completar los siguientes campos:

- **JUDGE\_ID:** Es el ID de nuestro usuario (El que recibimos por e-mail al registrarnos)
- **Language:** El lenguaje en el que está programada la solución (para nuestro caso: Java 1.7)
- **Problem:** El ID del problema a resolver
- **Solution source code:** Aquí podemos pegar el código de la solución (recomendado), o también existe la opción de seleccionar el archivo del código fuente. **LEER ANEXO I: REQUISITOS DEL CÓDIGO FUENTE.**

### Submit solution

Please [register](#) if you don't have JUDGE\_ID. Before submitting your first program look through the [guide](#).

JUDGE\_ID:  Language:  Problem:

Solution source code (not more than 64 KB):

Or solution source file (not more than 64 KB):

No se seleccionó un archivo.

## Paso 5: Ver resultados

Automáticamente luego de realizar un envío, se redireccionará a la URL con los resultados de todas las soluciones enviadas al juez online. Probablemente sea necesario recargar la página para que actualice el estado del envío.

Una forma manual de ir a la lista de resultados es acceder a ***Problems*** → ***Judge status***

## Anexo I: Requisitos del código fuente

Para qué el código fuente de la solución que enviamos pueda ser compilado y ejecutado por el juez online, debemos tener en cuenta las siguientes cuestiones:

- 1) El código no debe tener **package**.
- 2) Debe existir una única clase principal que debe tener el nombre **Main**. Si se requiere utilizar más de una clase se deben implementar **clases anidadas** (ver Anexo II).
- 3) La entrada y la salida del problema debe ser por consola **y no por archivos** (ver Anexo III).

## Anexo II: Clases anidadas

Para implementar clases anidadas en Java, es necesario que la clase principal genere un hilo de ejecución (thread) para que allí se ejecute el código principal de la solución, y desde allí si poder acceder a clases anidadas definidas dentro de la clase principal, como se muestra en el siguiente ejemplo:

```
public class Main {
    public static void main(string[] args) {
        new main().run();    // Genera el thread de ejecución
    }

    public void run() {
        // código principal del programa
        // Aquí si se pueden usar las clases anidadas
        ...
        ClaseAnidada1 objeto = new ClaseAnidada1();
        ...
    }

    public class ClaseAnidada1 {
        ...
    }

    public class ClaseAnidada2 {
        ...
    }
}
```



## Anexo III: Entrada y salida por consola

### Entrada:

Para la entrada es necesario tomar el flujo de entrada de la consola mediante un ***InputStreamReader***, pasándole como parámetro ***System.in***.

Tenemos dos opciones para la lectura desde consola:

#### Opción 1:

Utilizar ***BufferedReader*** y parsear los datos con ***split*** como cuando se lee de archivo:

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

String linea = in.readLine();
String[] valores = linea.split(" ");
```

#### Opción 2: (recomendada ya que es más eficiente en tiempo y memoria)

Utilizar ***StreamTokenizer*** para tomar los valores individualmente:

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

in.nextToken();
int entero = (int)in.nval;
in.nextToken();
int realDoble = in.nval;
in.nextToken();
String cadena = in.sval;
```

### Salida:

Se realiza mediante los métodos:

```
System.out.print();
System.out.println();
```

**Aviso:** La salida que entregue el programa debe ser exactamente como lo requiere el ***OUT*** del problema, ya que si contiene mensajes, mensajes de debugging, etc. que no correspondan a la salida, el juez online dará como resultados ***Wrong answer***.