

PROGRAMACIÓN COMPETITIVA

URJC - 2019



PROGRAMACIÓN COMPETITIVA

URJC - 2019

Organizadores:

- Jakub Jan luczyn (jj.luczyn@alumnos.urjc.es)
- Raul Martín (raul.martin@urjc.es)
- Ivan Martín (i.martinde.2016@alumnos.urjc.es)
- Isaac Lozano (isaac.lozano@urjc.es)
- Jesús Sánchez-Oro (jesus.sanchezoro@urjc.es)
- Juan David Quintana (juandavid.quintana@urjc.es)



Motivación

- Programming skills
 - diseño de algoritmos
 - estructuras de datos
 - nociones de complejidad
 - ...aprobar asignaturas!!! (ED, EDA, DAA, ...)
 - diversión



Motivación

- Empresas patrocinadoras
 - cazatalentos
 - concursos internos
 - entrevistas de trabajo



Motivación

- Participación en concursos
 - SWERC (Europeo del SurOeste)
 - AdaByron
 - Google Hashcode
 - 12 Uvas
 - ... premios y honor!!!





OBJETIVOS EN PROGRAMACIÓN COMPETITIVA

- Resolver los problemas en el menor tiempo posible
- Trabajo en equipo (nombres creativos. Ej. Calce Team, Raspu Team, etc)
- Representar tu institución, país...



PLANIFICACIÓN DE LA CHARLA

Objetivos:

- Familiarizarse con jueces
- Que aprendan nociones básicas de la programación
- Familiarizarse con los entornos de desarrollo
- Realizar un primer concurso de programación



TIPOS DE COMPETICIONES

- ACM-ICPC:
 - 5 horas de duración
 - Equipos: 3 personas (1 ordenador)
 - Puntuación: problemas resueltos (0/1)
 - Empates: tiempo + penalizaciones



TIPOS DE COMPETICIONES

RANK	TEAM	SCORE	A	B	C	D	E	F	G	H	I	J	K
1	 ENS Ulm 1 ENS Paris	10 1526	1/33	2/139	0	1/39	1/59	1/287	4/235	1/299	1/100	1/154	1/101
2	 Team RacIETH ETH Zürich	9 888	1/14	1/56	0	1/38	1/67	1/83	5/278	2/118	1/117	4	1/17
3	 UPC-1 Universitat Politècnica de Catalunya	9 1421	2/9	1/29	0	2/61	1/82	3/147	3/274	1/196	1/241	0	3/222
4	 SNS 1 Scuola Normale Superiore	8 803	1/13	2/70	1	4/113	1/23	1/107	0	2/147	1/152	0	1/78
5	 illUSIon Università della Svizzera italiana	8 986	1/9	2/103	0	2/66	1/80	2/157	0	1/249	2/206	1	1/36
6	 UPC-2 Universitat Politècnica de Catalunya	8 990	2/14	4/89	0	1/112	1/25	1/128	0	3/196	1/259	4	1/47
7	 EP red École Polytechnique	7 1092	1/17	2/210	0	5/123	1/79	1/240	0	0	2/166	0	1/137
8	 UPC-3 Universitat Politècnica de Catalunya	7 1168	1/12	1/231	0	1/97	1/59	2/170	0	0	5/282	0	2/197
9	 Moradonellani Politecnico di Milano	7 1181	1/16	1/102	0	2/50	3/78	5/285	0	0	2/223	0	1/267
10	 SNS 2 Scuola Normale Superiore	7 1319	3/69	1/175	0	2/107	2/140	1/236	0	2	1/295	0	2/197

<https://swerc.eu/2018/theme/scoreboard/public/>



TIPOS DE COMPETICIONES

- ACM-ICPC (Proceso de selección)
 - Eliminatorias en la universidad si hay más de tres equipos
 - Eliminatorias en el conjunto de países que forman una región (South-Western Europe)
 - Eliminatorias entre los potenciales candidatos en todo el continente (Super regional europeo (Beta))
 - Final Mundial



TIPOS DE COMPETICIONES

- Codeforces y Topcoder
 - Concursos muy rápidos y frecuentes
 - Libre para cualquiera
 - Dos o tres divisiones para novatos y expertos
 - De 95 a 120 minutos de duración
 - Puedes ver y ‘romper’ el código de otros
 - Sistema de puntuación (mientras más tardes en resolver problemas, más te penalizan en puntos)



TIPOS DE COMPETICIONES

- Facebook Hacker Cup y Google Code Jam
 - Evento de gente masiva online
 - Al menos 4 rondas
 - Suele haber ronda de clasificación, 2 rondas de filtro y luego la fase final
 - Dos tipos de evaluación (small y large)
 - El caso small se corrige automáticamente
 - El caso large se corrige al terminar la competición
 - Se permite cualquier tipo de solución (incluso manual ó *hardcodeada*) que permita llegar al output



TIPOS DE COMPETICIONES

- USACO/COCI/IOI
 - Concursos dirigidos a alumnos de bachiller/secundaria
 - ¡NO SON TAN FÁCILES!
 - Son evaluados con sistemas de puntuación (no binario ni penalizando tiempo de solución)
 - Resultados después de la competición
 - Funcionan por temporadas (de noviembre a abril) por ser eliminatorias para el IOI (International Olympiads in Informatics)



CARACTERÍSTICAS DE UN PROBLEMA

Enunciado: Se explica el problema con una narración que lo justifica

Análisis del Problema: Se requiere una solución determinista para el problema (siempre encontraremos una solución óptima y válida)

Entrada: Se especifica lo que nuestro programa debe leer

Salida: Se especifica lo que nuestro programa debe mostrar

Ejemplos I/O: Muestras de entrada/salida con el comportamiento esperado para el programa

Límites [Opcionales]: Lo máximo ó mínimo en cuanto a variables que nuestro programa debe tomar en cuenta



ABRIR MYAPPS URJC

The screenshot shows the myApps URJC web interface. At the top, there is a dark header bar with the Universidad Rey Juan Carlos logo and the myApps logo. The header also displays the user information: URJC \4eso3t1@alumnos.urjc.es, Cerrar sesión, and español (España, alfabetización internacional). Below the header, there is a navigation menu with tabs: Aplicaciones (selected), Descargas, my R:, and Ayuda. A search bar is located next to the tabs. On the far right of the header, there is a blue checkbox labeled "HTML5". Two red arrows point from the text "HTML5" in the previous slide to this checkbox.

Haga clic en el ícono de la aplicación que desea iniciar.

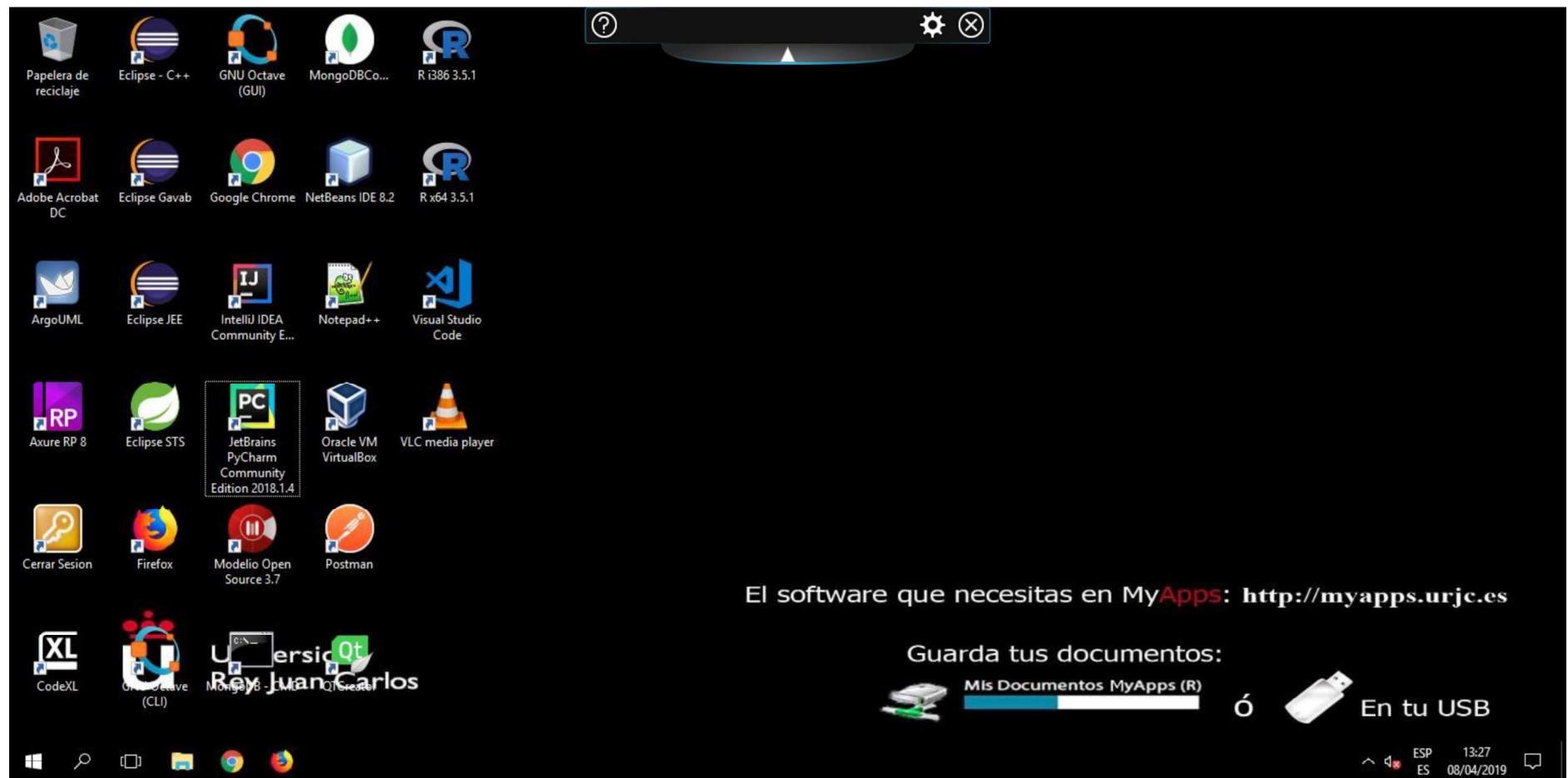
	Cerrar todas las Sesiones...		7- zip 18.01
	Explorador de Archivos myApps		Firefox
	Microsoft Access 2016		Microsoft Excel 2016
	Microsoft Publisher 2016		Microsoft Word 2016
			Adobe Acrobat DC
			Google Chrome
			Microsoft OneNote 2016
			Ubuntu
			Internet Explorer
			Microsoft PowerPoint 2016
			VLC Media Player
			Escritorio Desarrollo

Las aplicaciones aquí contenidas son exclusivamente para fines educativos, quedando expresamente prohibidos los usos comerciales, ilícitos o lesivos de bienes, servicios, derechos o intereses de terceros.

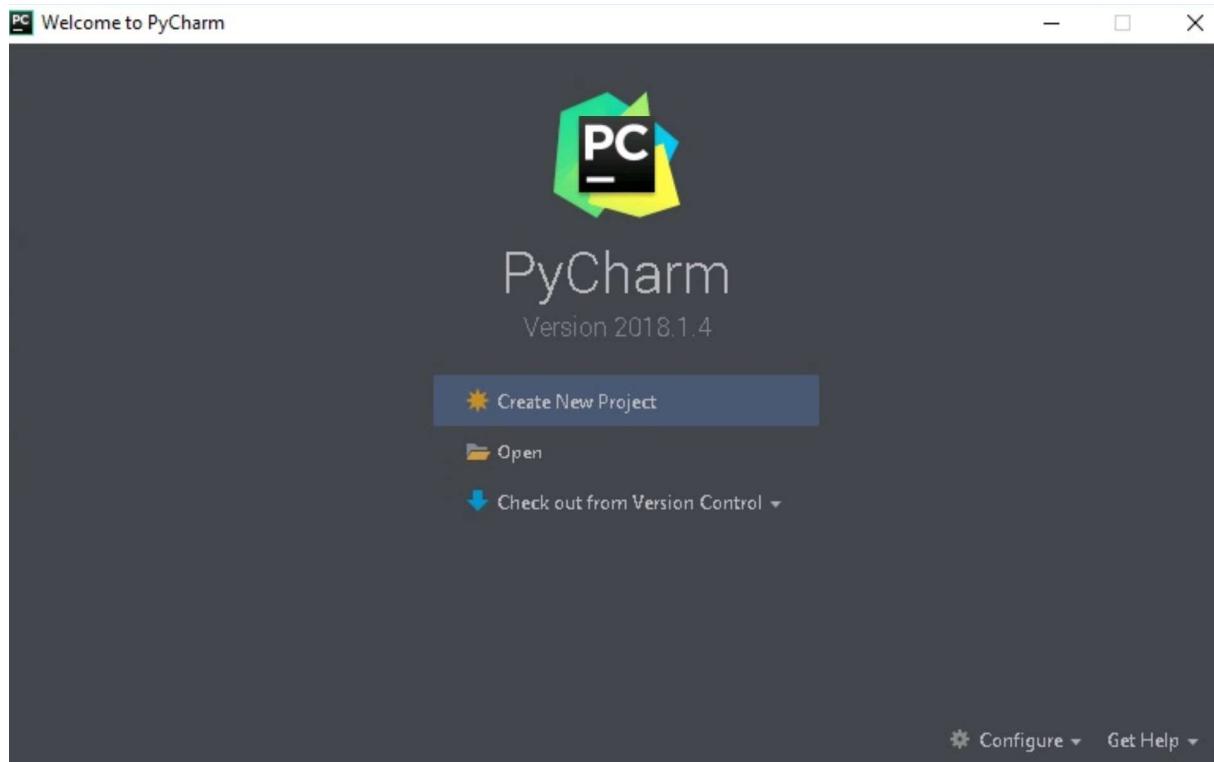
<https://myapps.urjc.es/myapps>



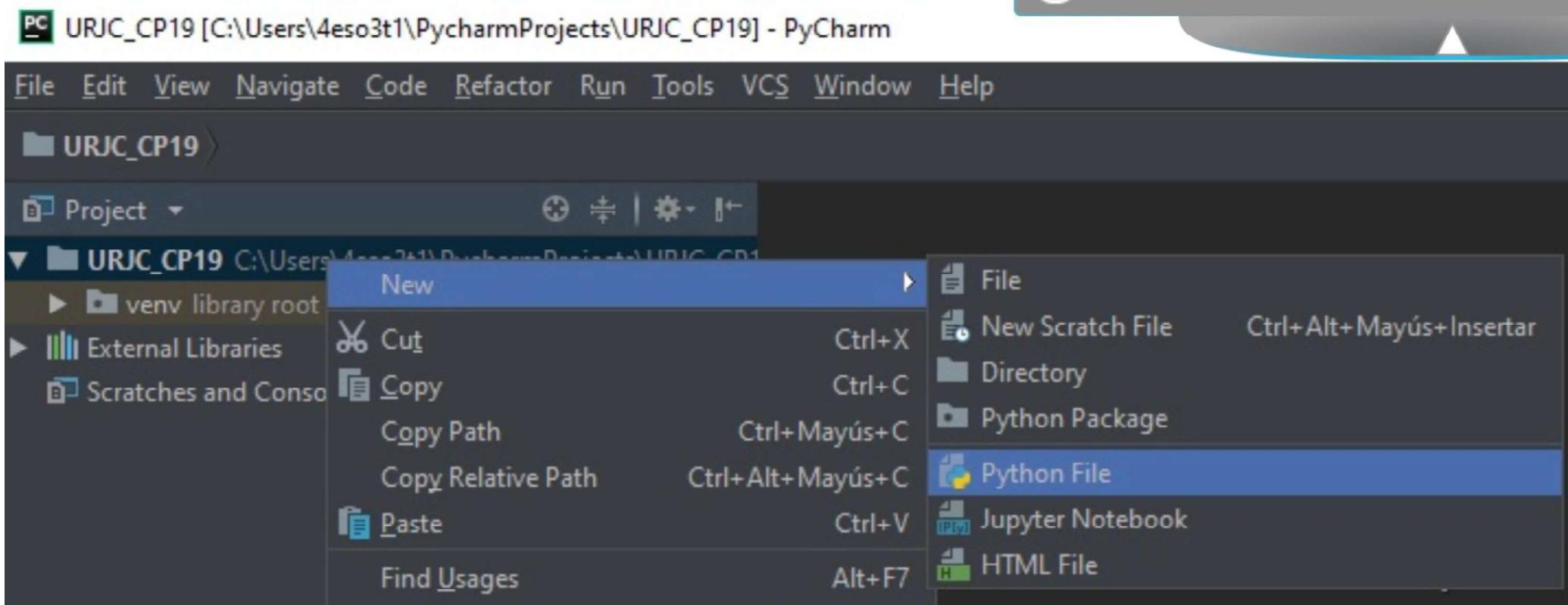
Escritorio de Desarrollo



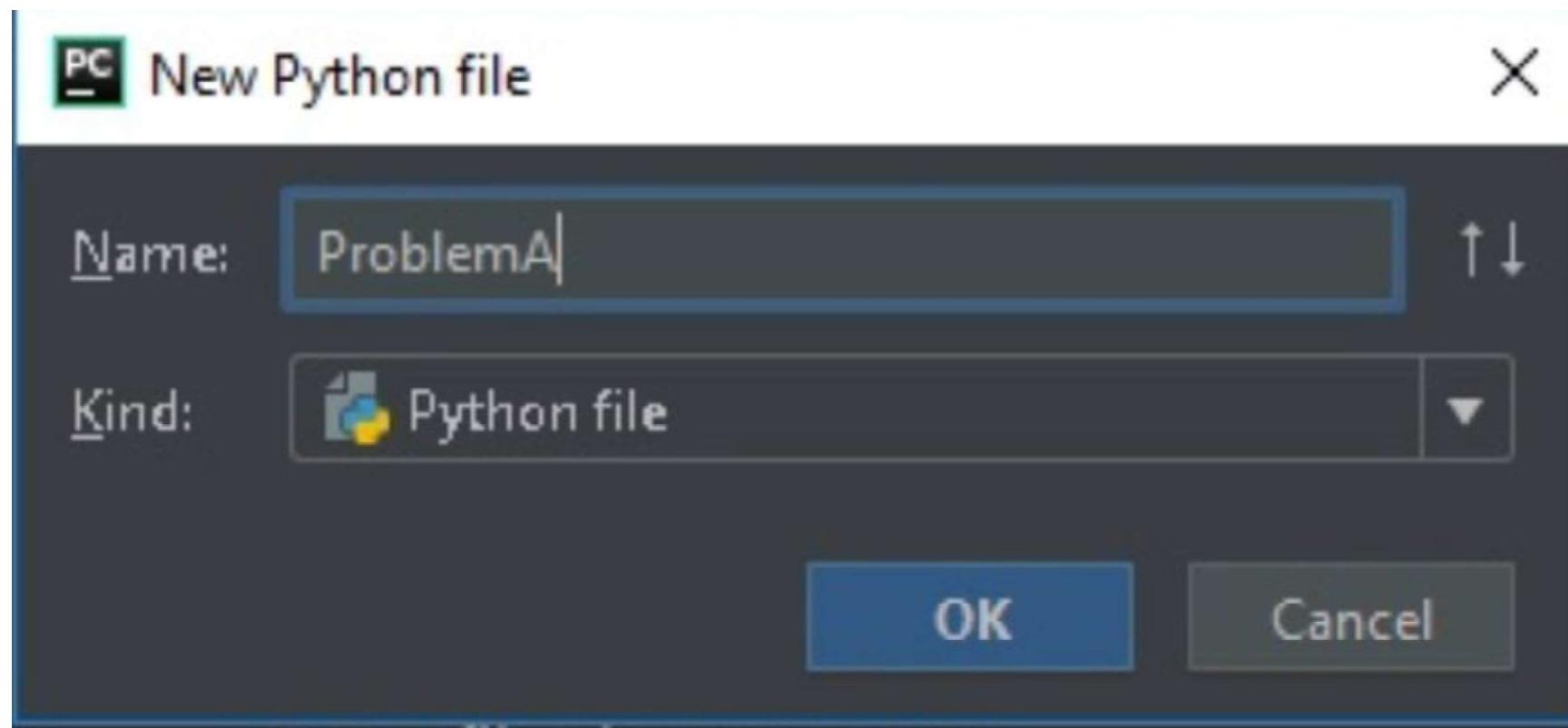
PyCharm - Crear Proyecto



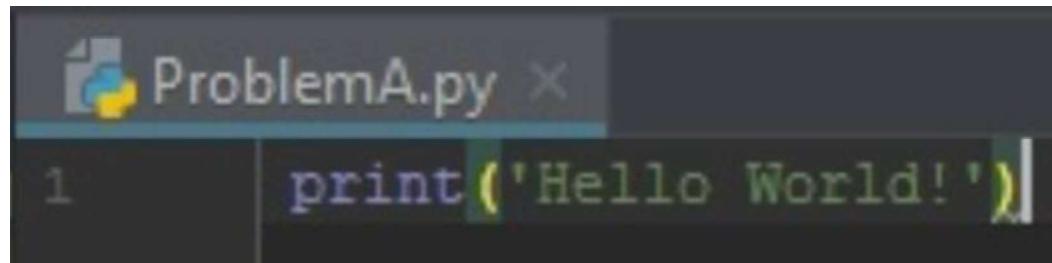
PyCharm - Crear Archivo .py



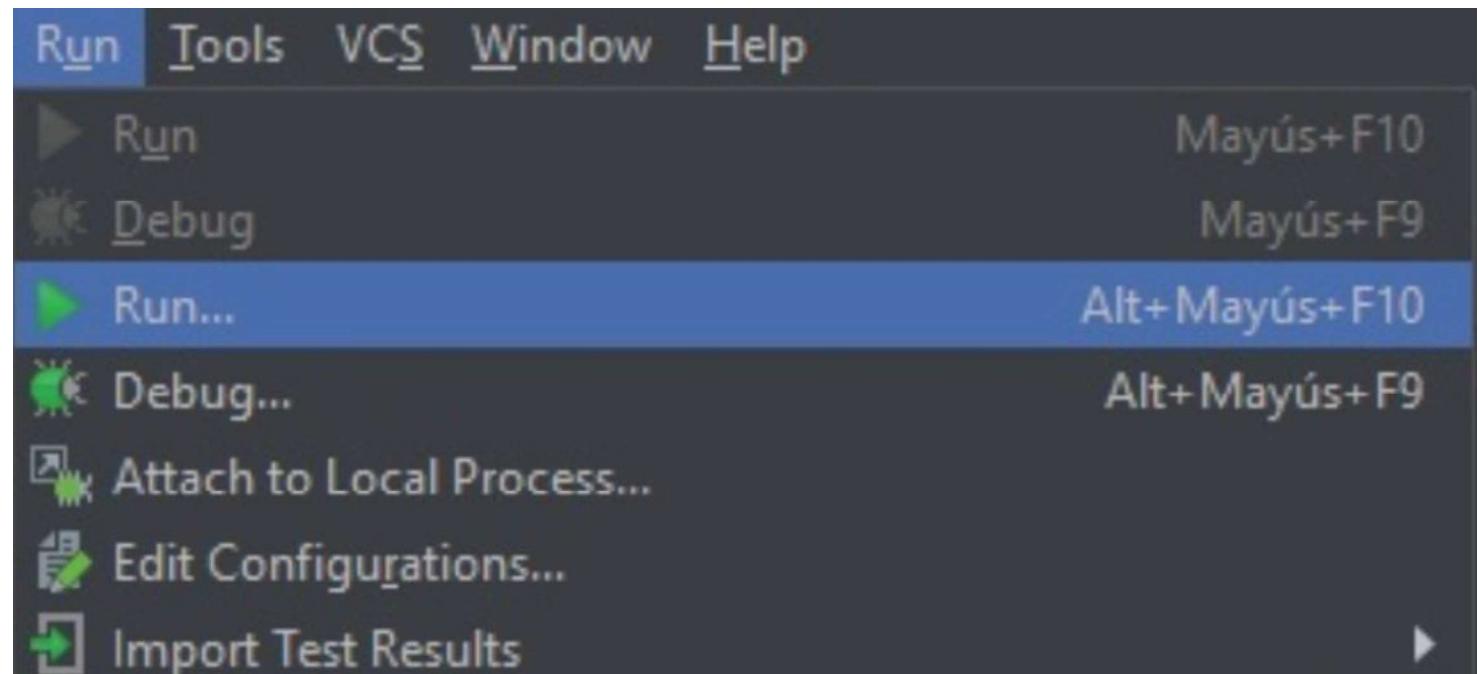
PyCharm - Crear Archivo .py



PyCharm - Primer Ejemplo!



A screenshot of the PyCharm IDE interface. At the top, there's a tab labeled "ProblemA.py X". Below the tabs, the code editor shows a single line of Python code: "print('Hello World!')".



PyCharm - Primer Ejemplo!

The screenshot shows the PyCharm IDE interface. The title bar reads "URJC_CP19 [C:\Users\4eso3t1\PycharmProjects\URJC_CP19] - ...\\ProblemA.py [URJC_CP19] - PyCharm". The menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The left sidebar shows the project structure with "URJC_CP19" expanded, containing "venv" and "library root", and "ProblemA.py" selected. The main editor window displays the Python code:

```
print('Hello World!')
```

. The bottom run tab shows the command: `C:\Users\4eso3t1\PycharmProjects\URJC_CP19\venv\Scripts\python.exe C:/Users/4eso3t1/PycharmProjects/URJC_CP19/ProblemA.py`, and the output: `Hello World!`.



LENGUAJES DE PROGRAMACIÓN

- Python



Sintaxis Básica Python

- Python
 - ¿Qué es una variable?

En programación una variable es un valor que puede cambiar dependiendo de las condiciones o la información que se le pasa al programa.

Por ejemplo si queremos tener en nuestro programa un número y utilizarlo lo podemos asociar a la variable y posteriormente operar sobre dicha variable.



Sintaxis Básica Python

- Python
 - ¿Qué es una variable?

```
num15 = 15
```

```
num15 = num15 + 5 # num15 valdrá 20
```

```
num15 = 2 # num15 valdrá 2
```

```
num10 = num15 # ambos valdrán 2
```

```
nombre = "Hola" # nombre es "Hola"
```

```
nombre = "Pepe" # nombre es "Pepe"
```



Sintaxis Básica Python

- Python
 - Pedir un valor: var = input()
 - Pedir un valor entero: var = int(input())
 - Pedir dos números, cada número estará en la variable correspondiente.
n1, n2 = map(int, input().split())



Sintaxis Básica Python

- Python

- Imprimir un salto de línea: `print ()`
- Imprimir texto sin salto de línea: `print (".", end="")`
- Imprimir un texto:

```
imprimir = "Texto"  
print(imprimir)
```

- Imprimir una variable

```
imprimir= "Texto" + str(var)  
print(imprimir)
```

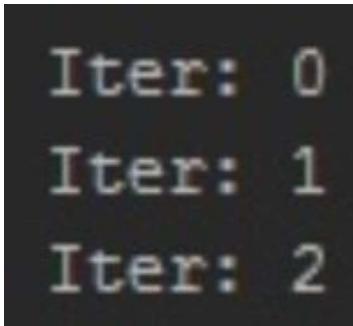


Sintaxis Básica Python

- Python
 - Hacer un bucle para recorrer N veces:

N = 3

```
for i in range(N) :  
    imprimir = "Iter: " + str(i)  
    print(imprimir)
```



```
Iter: 0  
Iter: 1  
Iter: 2
```

Tabulación!



Sintaxis Básica Python

- Python
 - Leer 4 palabras, por cantidad de palabras crear una variable:

```
p,p1,p2,p3 = input().split()
```

- Condiciones: Si numero es igual a 1 leer palabra, si no imprime la palabra

```
palabra = ""  
if(numero==1): //Si numero es igual a 1  
    palabra = input()  
else:  
    print(palabra)
```



Sintaxis Básica Python

- Python
 - Condicionales comparar palabras.

```
palabra = "NO"
if(palabra == "SI") :
    print("NO")
else:
    print("SI")
```

```
palabra = "SI"
if(palabra == "SI") :
    print("NO")
else:
    print("SI")
```



Sintaxis Básica Python

- Python
 - Operaciones matemáticas:

a = 1+1

a = a*1

b = 1+a

c = a-b

d = a//b #División de números enteros

e = 2*3



Sintaxis Básica Python

- Python
 - Comentarios en Python

Para comentar se utiliza el #, por ejemplo

```
print("Hola") #Imprime hola, soy un comentario
```

Para comentar múltiples líneas se utiliza """

```
#Todo comentado
""" print("Hola")
    #Imprime hola, soy un comentario
    print("Fin")
"""
```



Ejemplo Problema: ¡Hola mundo!

Problem description

You are to write the most basic program; it should just output “Hello world!” on a single line, no matter what the input.

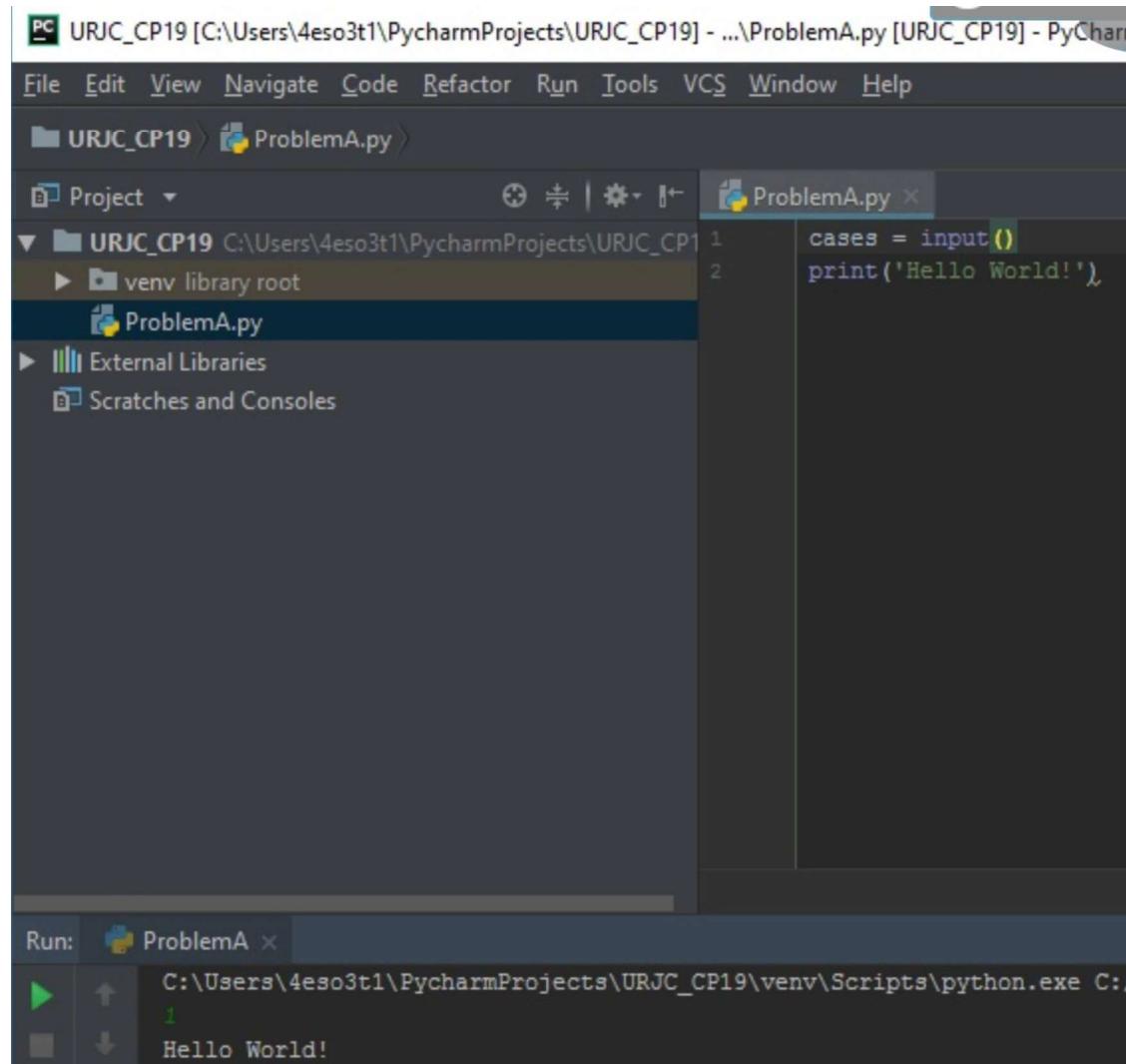
Sample input/output

Sample input and output for this problem:

Input	Output
1	Hello world!



Ejemplo Problema: ¡Hola mundo!



The screenshot shows the PyCharm IDE interface. The title bar reads "URJC_CP19 [C:\Users\4eso3t1\PycharmProjects\URJC_CP19] - ...\\ProblemA.py [URJC_CP19] - PyCharm". The menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The toolbar has icons for Project, Run, Stop, and others. The left sidebar shows the project structure with "URJC_CP19" containing "venv library root" and "ProblemA.py". The main editor window displays the code:

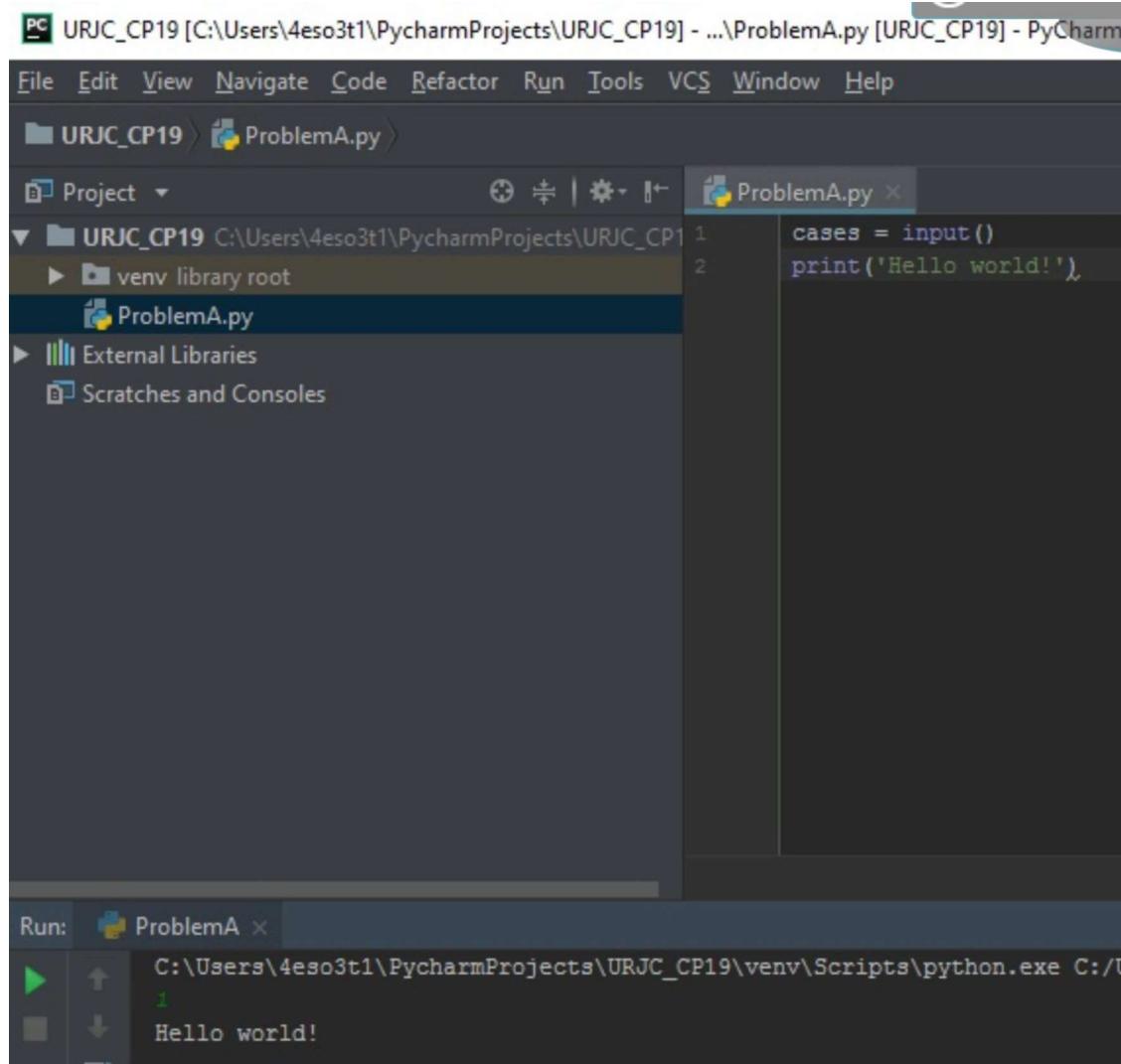
```
cases = input()
print('Hello World!')
```

The "cases" line has a red squiggly underline under the "cases" identifier, indicating a syntax error. The run tool window at the bottom shows the command "C:\Users\4eso3t1\PycharmProjects\URJC_CP19\venv\Scripts\python.exe C:/..." and the output "Hello World!".



Hello World! no
es lo mismo que
Hello world!

Ejemplo Problema: ¡Hola mundo!



The screenshot shows the PyCharm IDE interface. The title bar reads "URJC_CP19 [C:\Users\4eso3t1\PycharmProjects\URJC_CP19] - ...\\ProblemA.py [URJC_CP19] - PyCharm". The menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The toolbars have icons for Project, Run, and Stop. The left sidebar shows the "Project" view with a folder "URJC_CP19" containing "venv library root" and "ProblemA.py". The "External Libraries" and "Scratches and Consoles" sections are also visible. The main editor window displays the code "cases = input()\nprint('Hello world!')". The bottom "Run" tab shows the command "C:\Users\4eso3t1\PycharmProjects\URJC_CP19\venv\Scripts\python.exe C:/U" and the output "1\nHello world!".



Ejemplo Problema: ¡Hola mundo! N casos

Entrada

La primera línea contiene un entero N denotando un número entero.
La entrada debe ser leída de forma estándar.

Salida

Se debe imprimir la cadena “Hello World!” tantas veces como indique el número N .

La salida debe ser escrita de forma estándar.

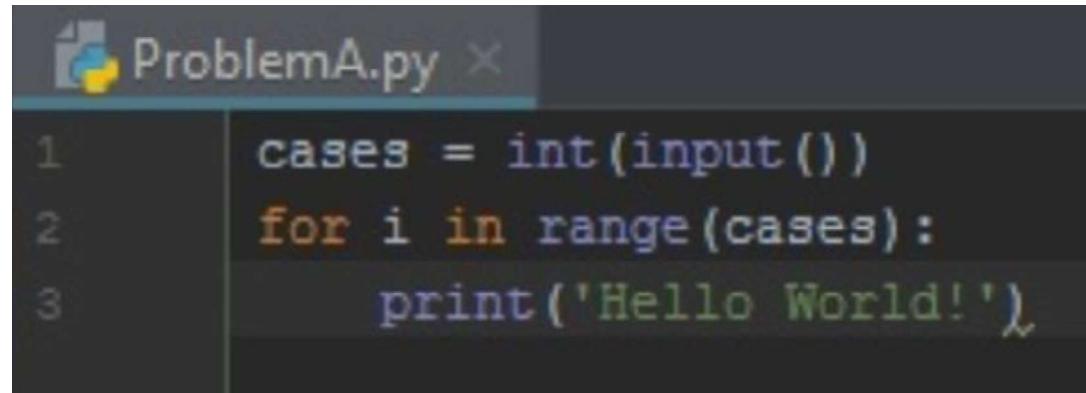
Entrada ejemplo	Salida ejemplo
5	Hello World! Hello World! Hello World! Hello World! Hello World!

Límites

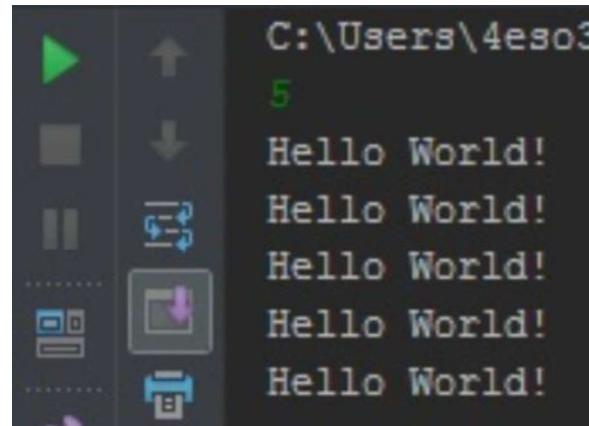
- $0 \leq N \leq 10$



Ejemplo Problema: ¡Hola mundo! N casos



```
ProblemA.py
1 cases = int(input())
2 for i in range(cases):
3     print('Hello World!')
```



```
C:\Users\4eso3
5
Hello World!
Hello World!
Hello World!
Hello World!
Hello World!
```

Importante la identación (tabulaciones) además de recoger el elemento como un entero.

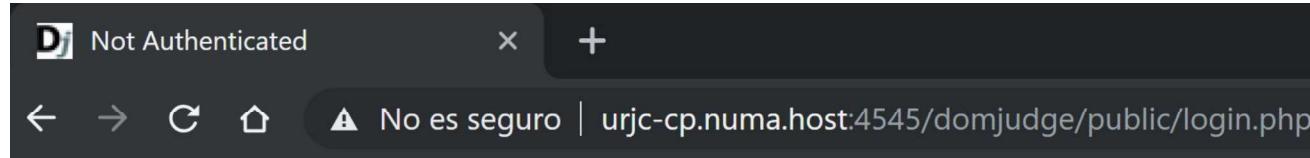
Domjudge

- Enlace:
<http://urjc-cp.numa.host:4545/domjudge/public/login.php>
- Nicks y contraseñas las tenéis en el papel



Domjudge

- Enlace: <http://urjc-cp.numa.host:4545/domjudge/>



Not Authenticated

Please supply your credentials below, or contact a staff member for assistance.

Login:

Password:

DOMjudge/5.3.2 at urjc-cp.numa.host Port 4545, page generated Thu 14 Mar 2019 18:56:34 CET



Domjudge

[overview](#) [problems](#) [scoreboard](#)

time left: 1d 4:56:05
contest: cp09mar
logged in as i.lozanoo x

RANK	TEAM	SCORE	A	B	C	D	E
1	i.lozanoo Universidad Rey Juan Carlos	5 10435	1/2051	2/2095	1/2056	2/2088	1/2105

Submissions

Elegir archivos Ningún archivo seleccionado problem language submit cancel

time problem lang result

2019- E CPP CORRECT

03-10

20:05

2019- B CPP CORRECT

03-10

19:55

2019- B CPP WRONG-ANSWER

03-10

19:54

2019- D CPP CORRECT

03-10

19:48

2019- D CPP WRONG-ANSWER

03-10

19:39

2019- C CPP CORRECT

03-10

19:16

2019- A CPP CORRECT

03-10

19:11

Clarifications

time from to subject text

2019- Jury All problem Para cada caso debes imprimir "NO" si no es posible recorrer
03-10 B todos los puentes u...
12:19

2019- Jury All problem Los límites efectivamente son estos últimos, ya están corregidos
03-10 B los enunciado...
19:41

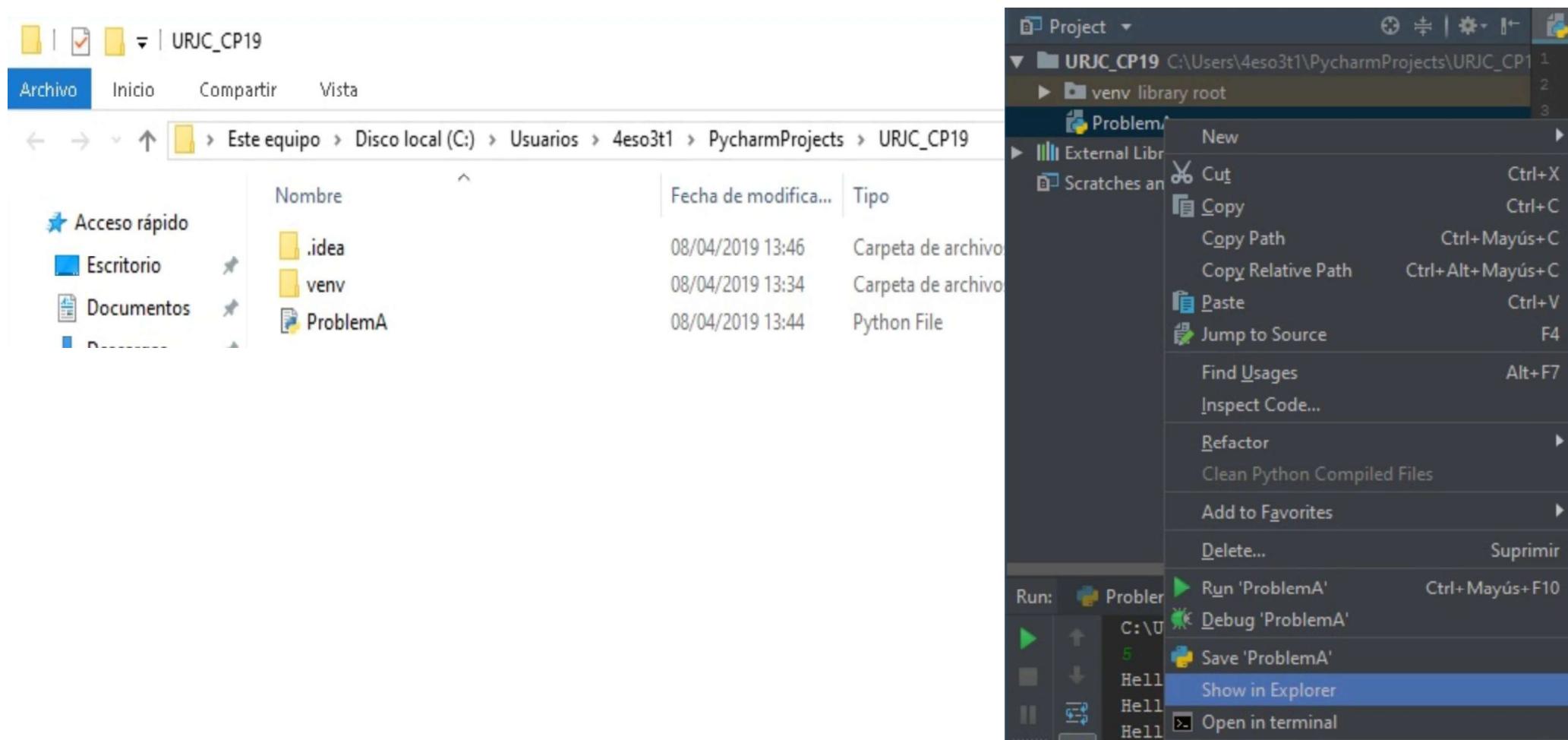
Clarification Requests

No clarification requests.

[request clarification](#)



Domjudge - Seleccionar Archivo



C:\Users\NOMBRE_DE_USUARIO\PycharmProjects\NOMBRE_DE_PROYECTO

Domjudge

[overview](#) [problems](#) [scoreboard](#)

Scoreboard Bloque 4 - Práctica

starts: 2019-03-09 09:00 - ends: 2019-03-15 23:55

RANK	TEAM	SCORE	A	B	C	D	E
1	i.lozanoo Universidad Rey Juan Carlos	5 10435	1/2051	2/2095	1/2056	2/2088	1/2105
2	j.noguerales Universidad Rey Juan Carlos	4 25824	0	2/6330	2/6457	4/6550	1/6387
3	a.garciaca.2016 Universidad Rey Juan Carlos	3 6805	1	7/2257	1/2102	10	5/2246
4	a.riveram.2016 Universidad Rey Juan Carlos	3 18330	0	6/5801	5/6432	1	3/5877
5	m.mardanyan.2016 Universidad Rey Juan Carlos	1 309	0	1/309	0	3	1
6	I.medina.2016 Universidad Rey Juan Carlos	1 5233	0	5/5153	0	0	0
7	j.sanchezfe.2018 Universidad Rey Juan Carlos	1 7824	0	3/7784	0	0	0



[!\[\]\(dcd6d77a18a931823dc16c9882eacfac_img.jpg\) overview](#) [!\[\]\(67e12f9967488cbd11833b81f12484dd_img.jpg\) problems](#) [!\[\]\(d018f6f0362f434cf560fecaebd71c4a_img.jpg\) scoreboard](#)

Contest problems

- **Problem A: Calabozo**
 [problem statement](#)
 [sample input](#) | [sample output](#)
- **Problem B: Puentes**
 [problem statement](#)
 [sample input](#) | [sample output](#)
- **Problem C: Espías**
 [problem statement](#)
 [sample input](#) | [sample output](#)
- **Problem D: Tareas**
 [problem statement](#)
 [sample input](#) | [sample output](#)
- **Problem E: Fiesta**
 [problem statement](#)
 [sample input](#) | [sample output](#)



¿Preguntas?

