

# UNCode

Plataforma online para la práctica frecuente y evaluación automática de tareas de programación de computadores, Notebooks de Python y lenguajes de descripción de hardware HDL

# UNCode

---

En este documento se presenta un resumen de las funcionalidades de la plataforma UNCode con propósitos informativos. Por lo tanto, este documento es opcional. Si está interesado en conocer información para usar la plataforma UNCode, por favor revise la **Guía de uso de UNCode**.

UNCode es una plataforma web educativa para aprender programación de computadores mediante su práctica frecuente. Soporta los siguientes lenguajes de programación: Python, Java, C/C++. Además, permite la evaluación automática de Notebooks de Python. También permite evaluar el diseño de circuitos digitales descritos en lenguajes de descripción de hardware – HDL.

UNCode realiza la evaluación automática ofreciendo al estudiante realimentación sumativa y formativa. La primera, permite conocer el resultado de forma inmediata e identificar posibles errores en sus soluciones. La segunda permite obtener información durante el proceso de desarrollo de las tareas que le permiten a los estudiantes construir conocimiento para saber cómo pueden proceder en caso de errores.

## Tareas de programación

La plataforma dispone de múltiples cursos en los cuales se asignan tareas de programación. Los estudiantes reciben un enunciado de un problema que los guía en el desarrollo de una posible solución, siguiendo los objetivos y requisitos específicos de la tarea. El flujo de trabajo general de UNCode funciona de la siguiente manera:

### Paso 1

Los estudiantes acceden a una tarea de programación particular en la plataforma, donde se presenta la descripción de la tarea.

### Paso 2

Desarrollan programas (código fuente o notebooks de Python), que constituyen las posibles soluciones para las tareas de programación propuestas

---

*Durante este proceso, los estudiantes pueden utilizar diferentes funcionalidades de apoyo que ofrecen una retroalimentación significativa (evaluación formativa) basada en su código fuente.*

---

### Paso 3

Los estudiantes envían su solución a la herramienta de evaluación automática. Esto puede hacerse varias veces sin penalidad en la calificación.

### Paso 4

Luego, UNCode informa la calificación obtenida, indicando la posible presencia de errores y los casos de prueba que el programa aprobó o reprobó (evaluación sumativa).

# UNCode

---

Para cada tarea el estudiante puede realizar uno o múltiples intentos (*submissions*) para subir una posible solución. Las tareas tienen propiedades que dependen de las características del problema, el historial de entregas del usuario y la naturaleza del ejercicio. Algunas de estas propiedades son:

## Fecha de entrega (*deadline*)

Las tareas asignadas tienen una fecha límite de entrega. A partir de la hora indicada, no se admiten más intentos.

## Estado (*status*)

Este estado define si el estudiante ha realizado o no intentos de resolución del ejercicio.

## Nota (*grade*)

Se indica el porcentaje de aciertos que tuvo el estudiante en la resolución del ejercicio (0 a 100%). La nota final corresponde al mejor intento histórico del estudiante.

En Restrepo-Calle et al. [1] se presenta una descripción exhaustiva de UNCode. Esta es una iniciativa del grupo de investigación [Programming Languages and Systems \(PLaS\)](#) de la Universidad Nacional de Colombia. Actualmente UNCode se encuentra desplegado en <https://uncode.unal.edu.co/>. Su código fuente está disponible públicamente en <https://github.com/JuezUN> y se distribuye bajo la licencia AGPL 3.0 (GNU Affero General Public License). UNCode está basado en el sistema de calificación automática conocido como [INGInious](#) [2]. Este sistema es flexible, seguro, escalable y extensible, y está orientado a entornos educativos, en lugar de concursos de programación competitiva.

# Referencias

---

- [1] F. Restrepo-Calle, J. J. Ramírez-Echeverry, and F. A. Gonzalez, “UNCode: Interactive System for Learning and Automatic Evaluation of Computer Programming Skills,” EDULEARN18 Proc., vol. 1, no. July, pp. 6888–6898, 2018. DOI: 10.21125/edulearn.2018.1632.
  
- [2] G. Derval, A. Gego, P. Reinbold, B. Frantzen, P. Van Roy, and U. De Louvain, “Automatic grading of programming exercises in a MOOC using the INGIInious platform,” Proc. Third Eur. MOOCs Stakehold. Summit, vol. 1, no. 12, pp. 86–91, 2015.

# Créditos

---

Facultad de

**I N G E N I E R Í A**

**Autor**

Felipe Restrepo Calle, PhD

**Asistente docente**

Alberto Nicolai Romero Martínez

**Diseño instruccional**

Claudia Patricia Rodríguez Sánchez

**Diseño gráfico**

Clara Valeria Suárez Caballero  
Milton R. Pachón Pinzón

**Fecha**

**Octubre 2020**

Facultad de  
**I N G E N I E R Í A**

Sede Bogotá



UNIVERSIDAD  
NACIONAL  
DE COLOMBIA

Facultad de  
**INGENIERÍA**  
Sede Bogotá



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA