



PRIMER PARCIAL
1 de septiembre de 2020

Indicaciones generales

- Este es un examen **individual** con una duración de **120 minutos: de 3:00 pm a 5:00 pm**.
 - En **e-aulas** puede acceder a las diapositivas y a la sección correspondiente a este parcial.
 - Solamente será posible tener acceso a **e-aulas.urosario.edu.co** y a los sitios web correspondientes a la documentación de Python dispuestos por el profesor, dentro de los que se incluye: **Turtle**.
 - Celulares y otros dispositivos electrónicos no deben ser utilizados durante el examen.
 - El estudiante solo podrá disponer de hojas en blanco como borrador de apuntes (opcional).
 - El estudiante puede tener una hoja manuscrita de resumen (opcional). Esta hoja debe estar marcada con nombre completo.
 - Como este parcial se realizará virtualmente, solo se considerarán las respuestas al parcial de aquellos estudiantes que se conecten a través de la plataforma, **enciendan su cámara y micrófono** durante la realización del parcial, y suban el parcial a e-aulas antes de retirarse del aula virtual.
 - La actividad en **e-aulas** se cerrará a la hora en punto acordada. La solución de la actividad debe ser subida antes de esta hora. El material entregado a través de **e-aulas** será calificado tal como está. Si ningún tipo de material es entregado por este medio, la nota de la evaluación será 0.0.
- Se aconseja subir a e-aulas versiones parciales de la solución a la actividad.**
- **Cualquier incumplimiento de lo anterior conlleva la anulación del examen.**
 - Las respuestas deben estar totalmente justificadas.
 - Use las plantillas **plantillaPY.Z**, con **Y = 1,2** y **Z = py, txt**, para dar su solución.
 - **Entrega:** tres archivos con extensión **' .py '** (Python) o **' .txt '** (Karel) según el caso, conteniendo el código solución, basado en las plantillas. Nombre los archivos como **pY.Z**, con **Y = 1,2** y **Z = py, txt**.
- Importante:** no use acentos ni deje espacios en los nombres de los archivos que cree.

1. En el siguiente reto Karel debe moverse a lo largo de un muro de forma indeterminada, como se muestra en la figura 1, de manera que el muro siempre esté a su derecha. Karel terminará de recorrer el muro en el momento en que se encuentre rodeado por muros como se observa en el estado final de la figura.
 - a) [35 ptos.] Escriba un programa en Karel que permita llevar a cabo esta tarea. Note las condiciones bajo las cuales Karel debe girar a la derecha o a la izquierda, al igual que la condición bajo la cual Karel termina su recorrido.
 - b) [15 ptos.] Al llegar a su destino, Karel debe depositar 50 beepers, como se observa en la figura 2. Suponga que Karel tiene en su maleta más de 50 beepers. Añada al programa las instrucciones necesarias para colocar los 50 beepers.

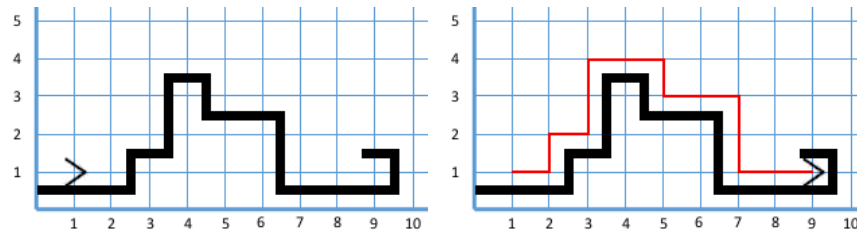


Figura 1: Estado inicial y estado final

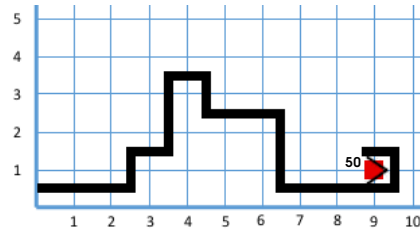


Figura 2: Estado final

2. Usando la librería de Turtle de Python realice los siguientes puntos:

- [35 ptos.] Defina una función `draw_poly (t, n, sz)` que hace que una tortuga `t` dibuje un polígono regular de `n` lados, donde cada arista tiene una longitud `sz`. Posteriormente utilizando un ciclo `FOR` invoque la función `draw_poly (t, n, sz)` para lograr que una instancia de Turtle llamada `manuelita` dibuje el conjunto de polígonos consecutivos de arista 30 como se muestran en la Figura 3. Los polígonos deben dibujarse con lápiz de color rojo y deben estar separados entre ellos 100 unidades.
- [15 ptos.] Finalmente, utilizando un ciclo `FOR` invoque la función `draw_poly (t, n, sz)` para lograr que una instancia de Turtle llamada `alex` dibuje un conjunto de polígonos consecutivos de arista 15 ubicándolos en las esquinas de un hexágono como se muestra en la Figura 4. En este caso los polígonos deben dibujarse con lápiz de color azul y deben estar separados entre ellos 200 unidades.

Utilice la plantilla adjunta para la realización del ejercicio.



Figura 3: Polígonos dispuestos en fila

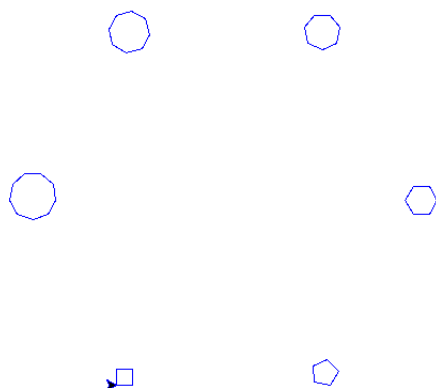


Figura 4: Polígonos dispuestos en las esquinas de un hexágono