**PY3**

**Listado de proyectos**

**Consideraciones:**

1. Para el proyecto 3 se tiene como criterio principal la libertad de elección es decir el grupo elige un tema libre. Sin embargo a modo de ayuda el proyecto considera la posibilidad de extender el proyecto2 considerando agregarle gráficos o sonidos o la posibilidad de agregar archivos o guardar archivos con la información generada. Adicionalmente los alumnos pueden elegir cualquiera de los proyectos que se indican en la lista que se presenta a continuación.

El alumno debe tener en su computadora o laptop instalado la versión 3.X de Python de 32 bits.

2. Debe instalar la biblioteca pygame y tkinter para python 3.x

Para instalar la versión de Python 3 de **pygame** usar la guía el repositorio <https://github.com/utec-cs1100/py2>

Para el uso de stddraw se incluye una guía en el repositorio <https://github.com/utec-cs1100/py2>

Tkinter viene instalado por default en las distribuciones de Python 3.x.

Ojo: Usar la versión 32 bits de python 3. Puede co-existir la versión 32 y 64 bits pero para el proyecto usar la versión 32bits.

3. El profesor proporciona las bibliotecas stddraw.py, color.py, stdaudio.py, stdio.py, picture.py, además de ejemplos de uso.

4. El profesor proporciona documentación de tkinter y ejemplos de uso.

5. Para los proyectos que requieran crear nuevos tipos de datos, almacenar y leer archivos (Programación orientada a objetos) utilizar la biblioteca dill. Esta se instala desde la carpeta <directorio donde está instalado el Python>/Scripts/ con el comando pip3 install dill

Luego se invoca con import dill, se les proporcionará ejemplos.

6. También es posible usar librerías gráficas tales como matplotlib, turtle, entre otros.

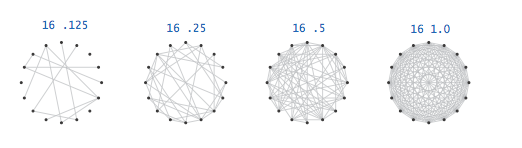
**Proyecto1**

Crear un generador de menú del día. Utilizando tkinter y Python usted creará una lista de menú con ingredientes y proceso de preparación para mostrarse en pantalla. Puede incluir imágenes en formato GIF o PNG. El usuario selecciona el día y usted elige aleatoriamente un menú para el día. En una semana no se puede repetir el plato. La información se almacenará en un archivo de la aplicación.

El programa incluye la creación de tipos de datos para el Menú y para las demás abstracciones que se consideren necesarias.

**Proyecto2**

Crear un programa que tome dos valores de la línea de comandos: Un entero N y un flotante p (entre 0 y 1) y dibuje N puntos separados de una circunferencia y con la probabilidad p dibuje líneas entre puntos de modo aleatorio que conectan dos puntos. Las líneas pueden tomar colores aleatorios.



**Proyecto3**

Teclado musical. Crear un teclado musical usando tkinter o stddraw y stdaudio para reproducir las notas musicales. Utilizar el ejemplo pro\_02.py. El teclado puede ser dibujado o usando botones de tkinter. Si se dibuja activar el sonido en las coordenadas que se elijan.

**Proyecto4**

El sistema solar. Crear un tipo de dato astro y organizar la información para almacenar sus datos tales como nombre, masa, tamaño y otras propiedades de los planetas que usted considere importante mostrar. El programa debe mostrar el sistema planetario, puede dibujar las líneas con picture y cargar imágenes en las posiciones correspondientes. Al momento de dar clic en uno de los planetas mostrar la información de ese planeta.

**Proyecto5**

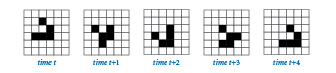
Juego de memoria. Crear un juego de memoria usando stddraw para mostrar un tablero de 5 x 5 figuras. El programa en primera instancia muestra todas las figuras ocultas. Cuando el usuario da clic en un recuadro se muestran las figuras correspondientes por un intervalo de tiempo, luego se vuelve a ocultar. Si el usuario da clic en una figura y luego en otra, se revelan las correspondientes. Usar la función picture y una matriz para almacenar las posiciones y el avance del jugador.

**Proyecto6**

El juego de la vida. Implementar un programa que simule el juego de la vida de Conway, considere una matriz de booleanos que corresponde a un sistema de células que pueden estar vivas o muertas. El juego consiste en verificar y actualizar el estado de cada celular dependiendo del estado de sus vecinos (las células adyacentes en toda dirección incluida diagonales). Una célula se mantendrá viva o muerta según las siguientes reglas:

* Una célula muerta con exactamente tres vecinos vivos revivirá.
* Una célula viva con exactamente un vecino vivo morirá.
* Una célula viva con más de tres vecinos vivos morirá.

Se puede inicializar con valores aleatorios o siguiendo los ejemplos de la imagen.



El programa debe solicitar un parámetro que indique el número de veces que se repetirá el ciclo y mostrar en un tiempo determinado cada ciclo de vida.