Unidade 6: Utilização das bibliotecas TI Hub & TI Rover | **Aplicação: Representação de um percurso**

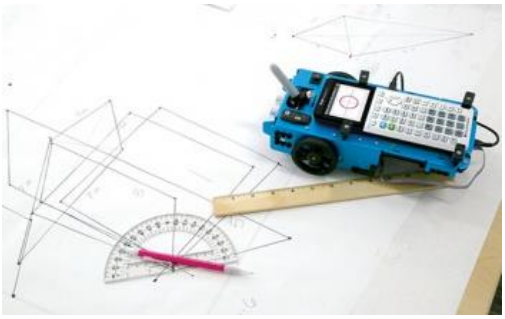
Nesta aplicação da Unidade 6 irá ligar o **TI-Innovator™ Rover** utilizando a biblioteca **TI Hub** e construir um programa para registar as coordenadas dos pontos durante um percurso e, de seguida, representá-los graficamente.

Objetivos:

- Explorar o módulo **TI Rover**.
- Escrever e utilizar um programa para usar o **TI-Innovator™ Rover** e os periféricos associados.
- Utilizar um ciclo fechado.
- Representar graficamente um conjunto de dados.

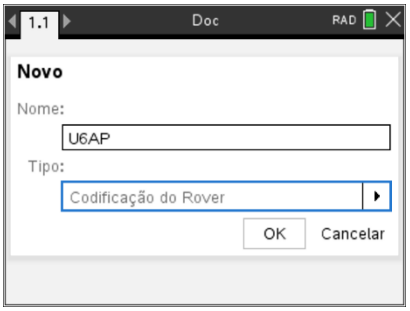
Nesta lição, iremos criar um programa que dará ao TI-Innovator™ Rover a capacidade de realizar um caminho correspondente ao desenho de um polígono regular.

As coordenadas dos vértices do polígono regular serão guardadas em listas e posteriormente representadas graficamente usando as instruções da biblioteca **TI Plot**.



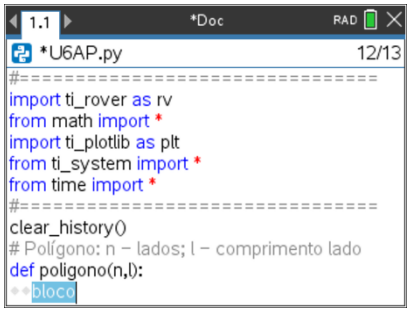
IMPLEMENTAÇÃO DO PROGRAMA:

- Inicie um novo programa no editor de TI-Python, designe-o por **U6AP**.



- Selecione tipo de programa **Codificação do Rover**, desta forma automaticamente as bibliotecas bibliotecas **TI Math**, **TI System** e **Time**, serão importadas. Também o módulo **TI PlotLib** será importado para o programa agora iniciado.

- Agora, poderá começar a escrever o programa.



- Embora possa não ser necessário, as instruções para limpar o ecrã e não mostrar o cursor são sempre mais agradáveis para o utilizador. Assim, colocaremos, sempre que nos parecer conveniente a instrução **clear_history()** localizada na biblioteca **TI System**.

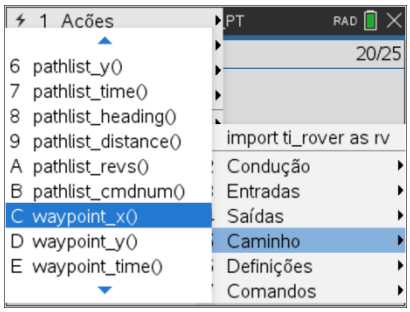
- Crie uma função **poligono()**, tomando como argumentos **n** o número de lados do polígono e **l** o comprimento, na unidade por defeito do TI-Innovator™ Rover, ou seja, o dm.

- A medida da amplitude do ângulo externo de cada polígono será, portanto, igual **a** = $\frac{360}{n}$. Porquê o ângulo externo?

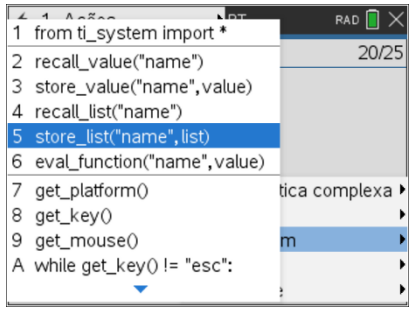
- Crie duas listas vazias, **abcissas** e **ordenadas**, destinadas a guardar as coordenadas de cada um dos **n** vértices.

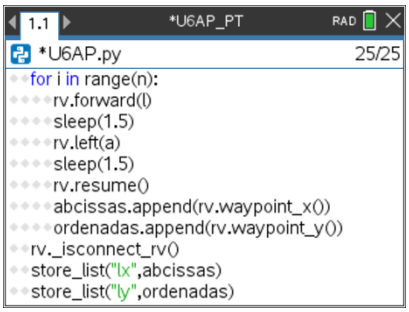
- Crie um ciclo aberto, **While**, de tamanho **n**.

- As instruções que se seguem, e que permitirão que o **Rover** percorra o percurso desenhado pelo polígono considerado, podem ser obtidas na biblioteca **TI Rover**. Estas instruções repetir-se-ão **n** vezes, tamanho do ciclo **While**.



  - Mover para a frente o **Rover** em **l** decímetros – **rv.foward(l)**.
  - Virar à esquerda com um ângulo de amplitude **a** – **rv.left(a)**.
  - Colocar, entre cada etapa, uma espera de 1,5s – **sleep(1.5)**.
  - Obter e guardar as coordenadas dos vértices nas listas **abcissas** e **ordenadas**. – **abcissas.append(rv.waypoint_x())** e **ordenadas.append(rv.waypoint_x())** .
  - Desconectar o **Rover** no final do percurso – **rv._isconnect_rv()**.
  - Exportar os dados das listas **abcissas** e **ordenadas**, respetivamente, para as listas **lx** e **ly**, possibilitando a representação gráfica e/ou exploração dos dados numa outra aplicação da TI-Nspire™ CX II (Gráficos, Listas e Folha de Cálculo, …).





OBSERVAÇÕES:

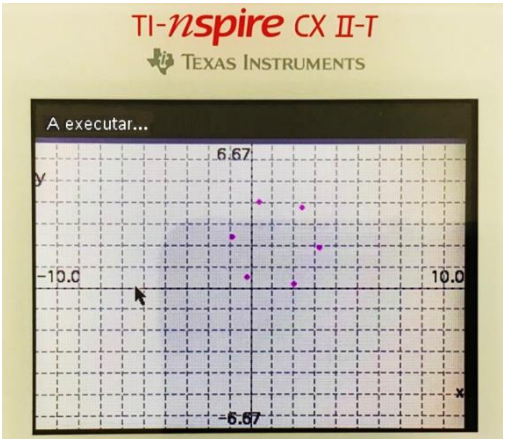
  - As instruções **rv.waypoint_x()** e **rv.waypoint_y()** encontram-se no módulo **TI Rover**, na opção **5: Caminho**, e de seguida a opção **C: waypoint_x()** .
  - A instrução **rv._isconnect_rv()** deve ser escrita a partir do teclado.
  - A instrução **store_list()** encontra-se no módulo **TI System**, na opção **5: store_list("name",list)**.

- Terminada a recolha de dados e o deslocamento do Rover, passemos para a representação gráfica dos dados. Podemos incluir este código na função já criada, ou, se for vantajoso, criar no mesmo programa uma nova função para a representação gráfica, como se procedeu em alguns exemplos das lições anteriores (Unidade 5).

- Todas as instruções usadas para a representação gráfica dos dados estão disponíveis no módulo **TI PlotLib**, que obrigatoriamente terá a sua biblioteca importada, nas opções **2: Configurar** e **3: Desenhar**.

EXECUÇÃO DO PROGRAMA:

- Execute o seu programa, atalho ctrl + R, e, no interpretador, execute a função **poligono()** para, por exemplo, percorrer e desenhar um quadrado com o lado de comprimento 1 dm: **poligono(4,1)**.



- Continue a testar o programa, agora com um hexágono regular com 2 dm de lado, **poligono(6,2)**, obtendo a seguinte representação.

OBSERVAÇÃO:

Para este tipo de exercício, evite utilizar as instruções **rv.pathlist_x()** e **rv.pathlist_y()**.
Pois, ao desenhar um segmento, serão registadas as coordenadas dos pontos de um segmento do polígono e, em seguida, no segmento seguinte serão registados novamente as coordenadas do último ponto do segmento anterior como o primeiro ponto do atual segmento.
Mais ainda, porque no nosso código colocamos um atraso de 1.5 s entre cada movimento do Rover. Portanto, as instruções **rv.path…** são, para o nosso caso, inapropriadas.
Usando as instruções **rv.pathlist_x()** e **rv.pathlist_y()**, obteríamos as coordenadas do ponto final duas vezes.

NOTE QUE:

Deve ajustar o formato de sua grelha de fundo, dependendo dos polígonos que deseja desenhar. Isso foi intencionalmente definido aqui nas configurações por defeito, a fim de observar a precisão da pista do TI-Innovator™ Rover.

Também deve ter em atenção a natureza da superfície na qual o Rover se irá deslocar. O piso não deve oferecer muita resistência ao movimento ou, pelo contrário, favorecer o deslizamento.