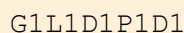
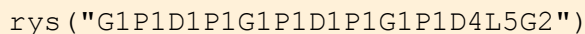
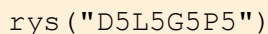


## Treść zadania

G1P2G1P2D2



*Przykłady:*



W każdym kroku iteracji ustawiamy kierunek żółwia na wskazany przez pierwszy element pary, przy czym kierunek 0 to prawo, a kąty są liczone przeciwnie do ruchu wskazówek zegara: G – 90, D – 270, L – 180, P – 0. Wyznaczamy także długość odcinka, który będzie rysowany. Gdy liczba jest



## Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

jednocyfrowa to mnożymy podaną liczbę przez 10. W przypadku liczb dwucyfrowych mnożymy liczbę dziesiątek przez 10 i dodajemy liczbę jednostki. Tak uzyskaną liczbę mnożymy przez 10.

Ogólna postać funkcji `rys(kod)` będzie wyglądać następująco:

```
dopóki istnieją nieprzetworzone znaki parametru kod:  
    weź pierwszy znak i przypisz go zmiennej kierunek  
    jeżeli kierunek == 'G', to ustaw kierunek żółwia na 90 stopni  
    jeżeli kierunek == 'D', to ustaw kierunek żółwia na 270 stopni  
    jeżeli kierunek == 'L', to ustaw kierunek żółwia na 180 stopni  
    jeżeli kierunek == 'P', to ustaw kierunek żółwia na 0 stopni  
    weź drugi znak i sprawdź, czy kolejny jest cyfrą  
    jeżeli tak, to zmiennej dystans przypisz wartość liczbową drugiego  
        znaku pomnożoną przez 10 i dodaj wartość liczbową trzeciego  
    w przeciwnym przypadku zmiennej dystans przypisz tylko wartość  
        liczbową drugiego znaku  
    narysuj odcinek o długości  $10 * \textit{dystans}$ 
```

### Rozwiązanie w języku Python

Trzon rozwiązania stanowi pętla `while` ze zmienną sterującą `i`, która najpierw przyjmuje wartość 0, a w każdym obrocie pętli jest zwiększana o 2 (linia 22) lub 3 (linia 20 i 22), gdy *dystans* jest opisany liczbą dwucyfrową. Zarówno przy kolejnej iteracji (linia 5), jak i przy sprawdzaniu, czy liczba jest dwucyfrowa (linia 18), trzeba badać, czy nie doszliśmy do końca zmiennej `kod`.

Przy implementacji zostały wykorzystane wbudowane funkcje: metoda `isdigit()`, która sprawdza czy znak jest cyfrą oraz z biblioteki `turtle` `seth(kąt)` – ustawienie żółwia w określonym kierunku.

Można też zamiast badać czy znak jest cyfrą za pomocą metody `isdigit()`, sprawdzać czy jest większy lub równy od znaku "0" i mniejszy lub równy od znaku "9".

```
1  from turtle import *  
2  
3  def rys(kod):  
4      i = 0  
5      while i < len(kod):  
6          #kierunek  
7          kierunek = kod[i]  
8          if kierunek == "G":  
9              seth(90)  
10         if kierunek == "D":  
11             seth(270)  
12         if kierunek == "L":  
13             seth(180)  
14         if kierunek == "P":  
15             seth(0)  
16         #dystans  
17         dystans = int(kod[i+1])  
18         if i + 2 < len(kod) and kod[i + 2].isdigit():  
19             dystans = 10 * dystans + int(kod[i + 2])  
20             i += 1  
21         fd(10*dystans)  
22         i += 2
```



## Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

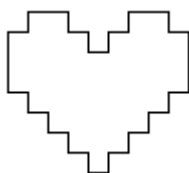
### Testy

Wywołujemy funkcję `rys()` dla różnych parametrów. Należy sprawdzić, czy żółw prawidłowo rozkodowuje wszystkie cztery kierunki oraz czy funkcja prawidłowo interpretuje liczby jednocyfrowe i dwucyfrowe opisujące dystans, o który przemieszcza się żółw.

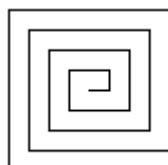
Ponieważ w Pythonie można korzystać z operatora mnożenia parametry funkcji zapisane są skrótowo, np. `"D1" * 5` oznacza pięć powtórzeń napisu `"D1"`, czyli napis `"D1D1D1D1D1"`.

Gdy chcemy przyspieszyć tworzenie rysunku przez żółwia, stosujemy wywołanie złożone z funkcji `tracer()` – rysownie w pamięci, właściwego wywołania funkcji `rys()` i na końcu uaktualniamy ekran za pomocą funkcji `update()`.

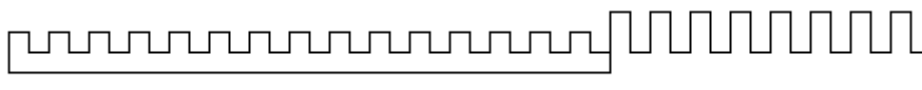
```
tracer(0); rys("D5L5G5P5"); update()
```



```
rys("G3P1G1P2D1P1D1P1G1P1G1P2D1P1D3L1D1L1D1L1D1L1G1L1G1L1G1L1G1L1")
```



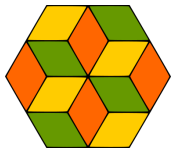
```
rys("P1G1L2D2P3G3L4D4P5G5L6D6P7G7L8D8")
```



```
rys("D2L30G1"+"G1P1D1P1"*15+"G2P1D2P1"*8+"D2"+"L46")
```



```
rys("L10"+"L20"+"D1"*5+"P1"+"G4"+"P2"*18+"D2"*2+"P1"+"G5"+"L5")
```



## Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

---



rys("D30G60P2D12L1D47L1")