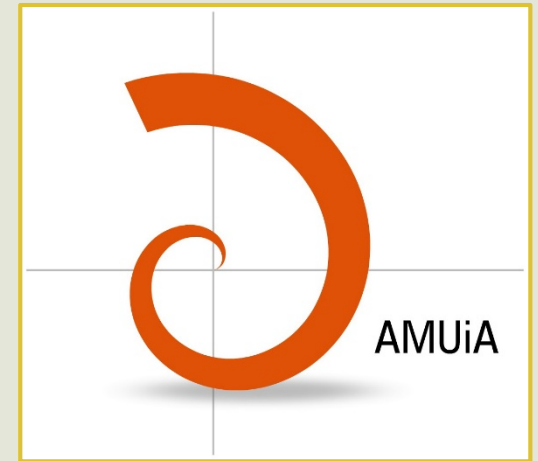


Jakub Jernajczyk – *artysta grafik, matematyk*
Bartłomiej Skowron – *filozof, matematyk*
Jarosław Drapała – *informatyk, statystyk*
+
Radosław Rudek – *informatyk*



Język maszyn

Paradygmat Rybczyńskiego



- Otacza nas coraz więcej **maszyn**, chcemy umieć z nich **świadomie** korzystać
- Maszyna powinna być **narzędziem**
- Wszystko co pokazuje, odtwarza i wykonuje maszyna, zostało **napisane** w języku tej maszyny (zaprogramowane)
- Nie jestem analfabetą, ponieważ umiem czytać, pisać i ... **programować**

Cele główne projektu **język maszyn**

- **Umasowienie** umiejętności programowania
- Postawienie maszyny w roli narzędzia używanego przez:
 - artystów,
 - inżynierów,
 - badaczy,
 - myślicieli,
 - uczniów,
 - pracowników,
 - ...

Założenia projektu **język maszyn**

- Programowania jest w stanie nauczyć się **każdy**
- Nauka programowania nie wymaga wysokich nakładów finansowych
- Wszystkie języki programowania opierają się na kilku podstawowych konstrukcjach programistycznych
- Ucząc się podstaw programowania, uczymy się **sztuki rozwiązywania problemów**
- Istniejące narzędzia dydaktyczne skierowane są do odbiorcy, który ma pewne przygotowanie matematyczne
- Podział na „humanistów” i „ściśłowców” jest sztucznym wytworem społecznym
- Chcemy dotrzeć do osób o różnych typach umysłowości i zdolności

Co **nie jest celem** projektu język maszyn

- Nie chcemy, aby wszyscy byli programistami
- Nie jest naszym zamiarem kształcenie rzeszy programistów na potrzeby rynku pracy (choć nie ukrywamy, że projekt da się w podobnym celu wykorzystać)
- Nie chcemy zastąpić zajęć szkolnych z informatyki

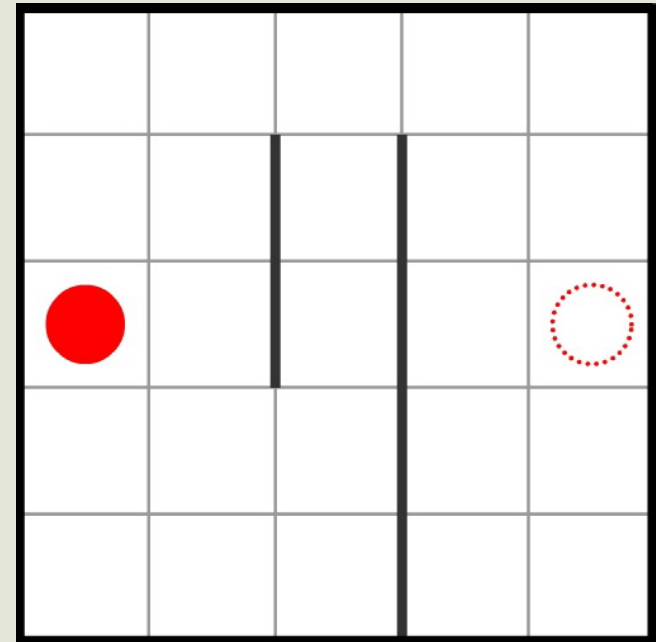
Metody

- Korzystanie z pewnych elementów bogatego dorobku polskiej dydaktyki informatyki
- Oparcie programu nauczania o fundamentalne zdolności percepcyjne człowieka
 - widzenie
 - myślenie wzrokowe
 - ? dźwięk ?
- Przygotowanie przykładów z różnych dziedzin sztuki, nauki i techniki



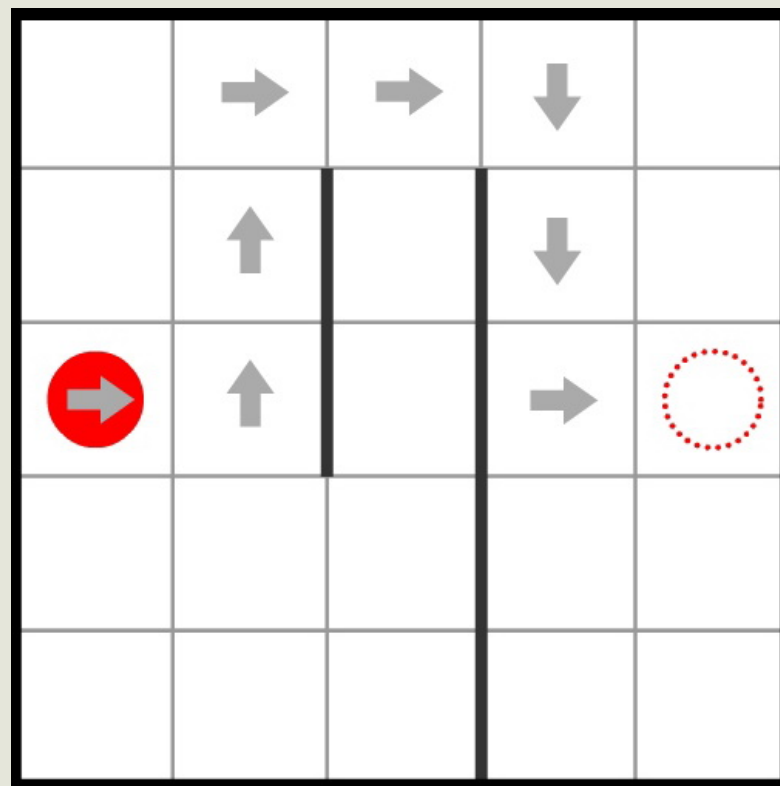
Metody – graficzny język programowania

- Cel nauczania : nabycie intuicji podstawowych pojęć:
 - program
 - powtórzenia / pętla
 - paczka (zmienna)
 - instrukcja warunkowa
- Narzędzia dydaktyczne
 - scena z kulką i przeszkodami
 - język programowania oparty o piktogramy
 - uporządkowany zestaw ćwiczeń



Metody – graficzny język programowania

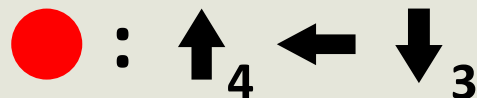
- Program



● : → ↑ ↑ → → ↓ ↓ →

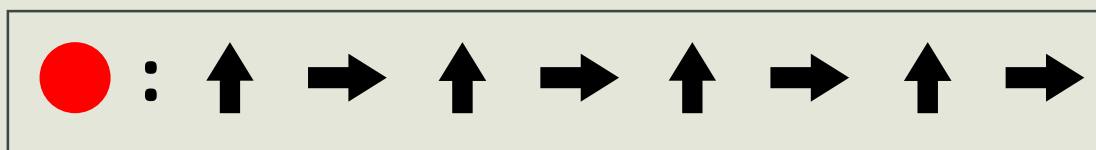
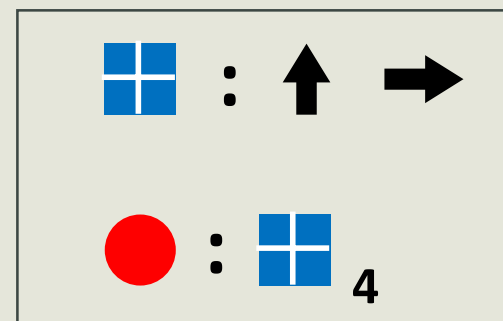
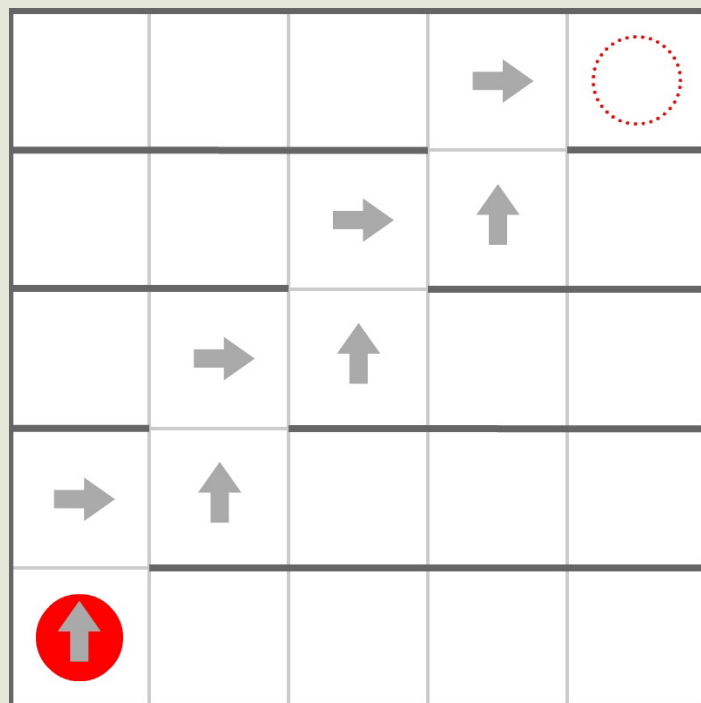
Metody – graficzny język programowania

- Powtórzenia / pętla



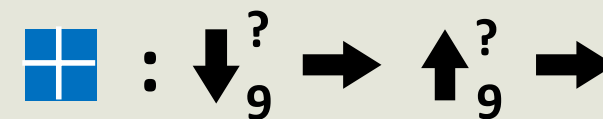
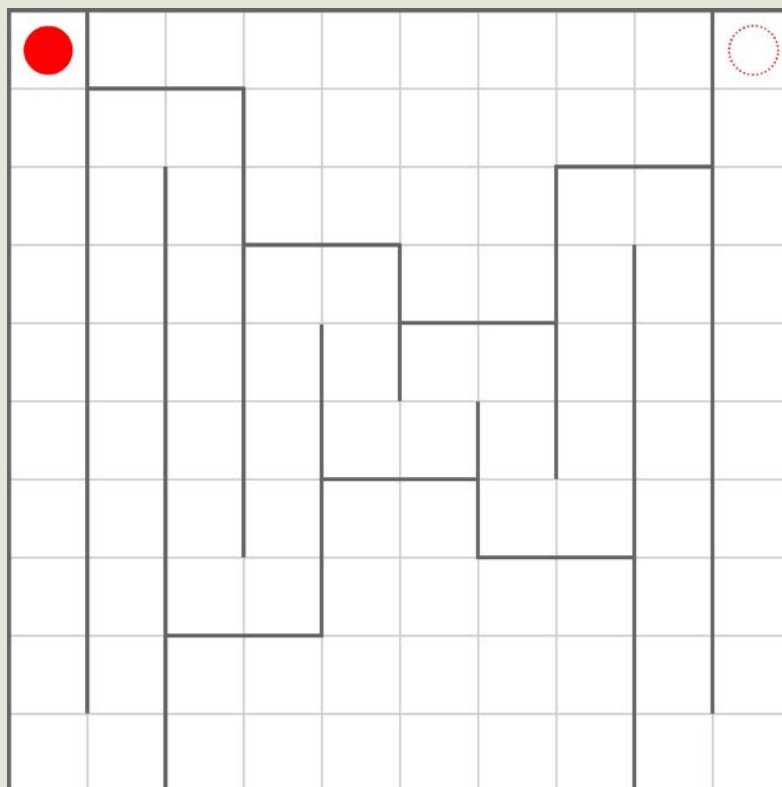
Metody – graficzny język programowania

■ Paczka



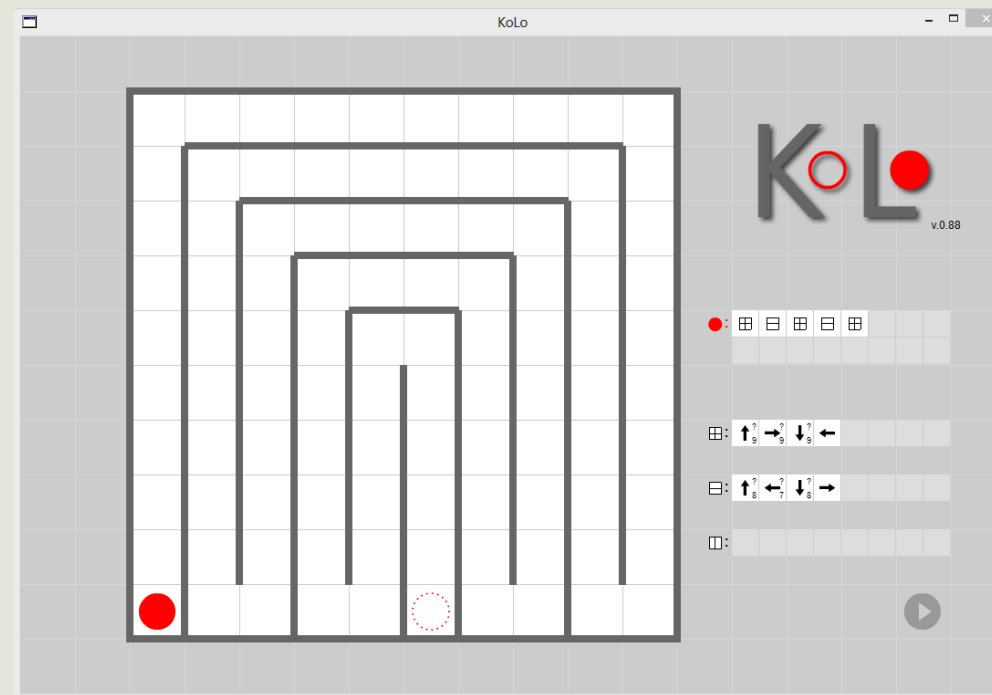
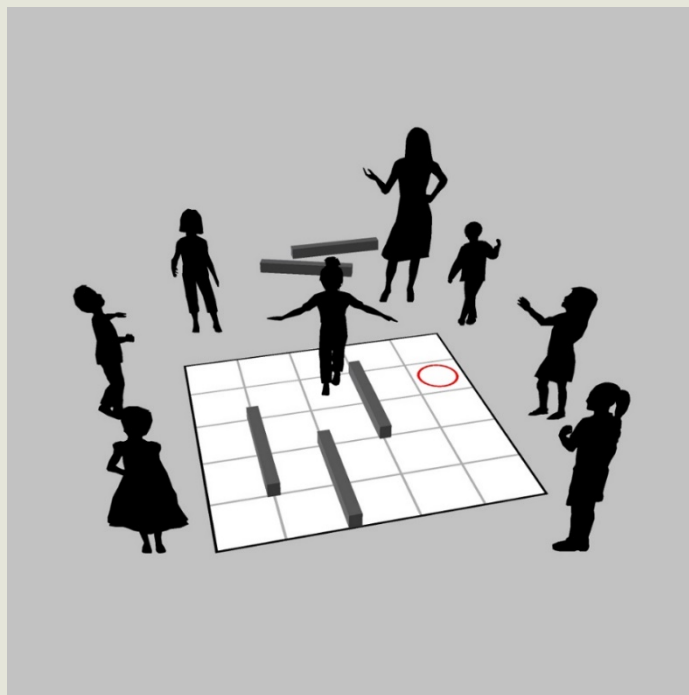
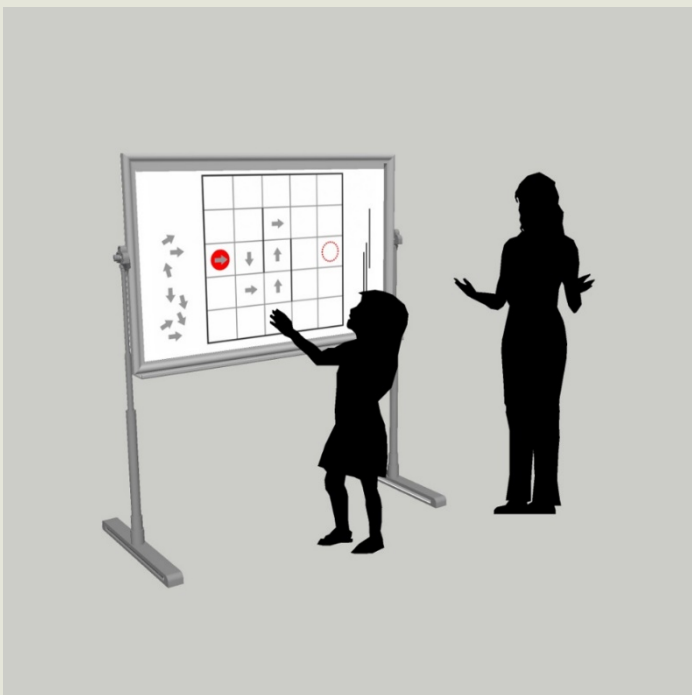
Metody – graficzny język programowania

- Instrukcja warunkowa



Metody – graficzny język programowania

- Realizacja praktyczna



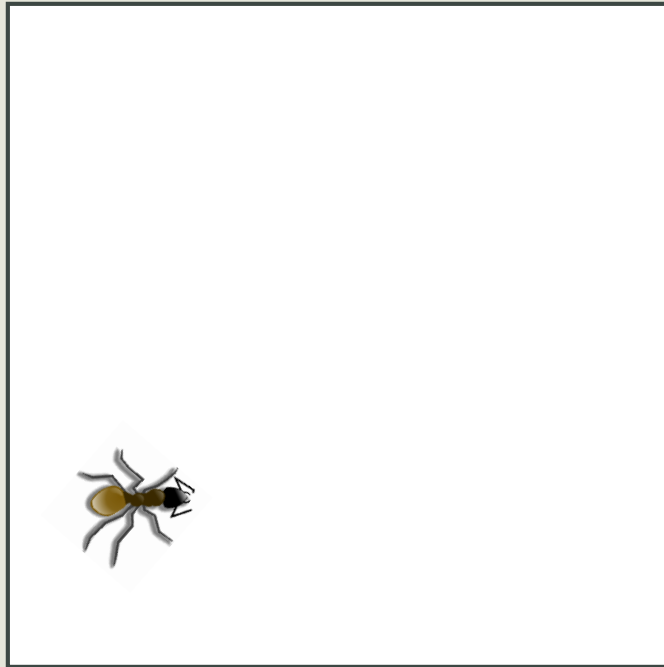
Metody – Octave + mrówkaGo

- Inspiracja dla mrówki : LOGO
- Powody wybrania Octave:
 - **darmowe** środowisko programowania
 - język programowania identyczny z **MATLAB-em**
 - programy mają wyłącznie postać **plików tekstowych**
 - można programować **bez wdawania się w szczegóły techniczne**
 - najważniejsze: **krótki czas między pisaniem programu a efektem końcowym**
- Cel mrówki: zapewnić stopniowe przejście **od zabawy do prawdziwego programowania**

Metody – Octave + mrowkaGo

- **f** – idź do przodu jeden krok

'fff'



Metody – Octave + mrowkaGo

- **f** – idź do przodu jeden krok

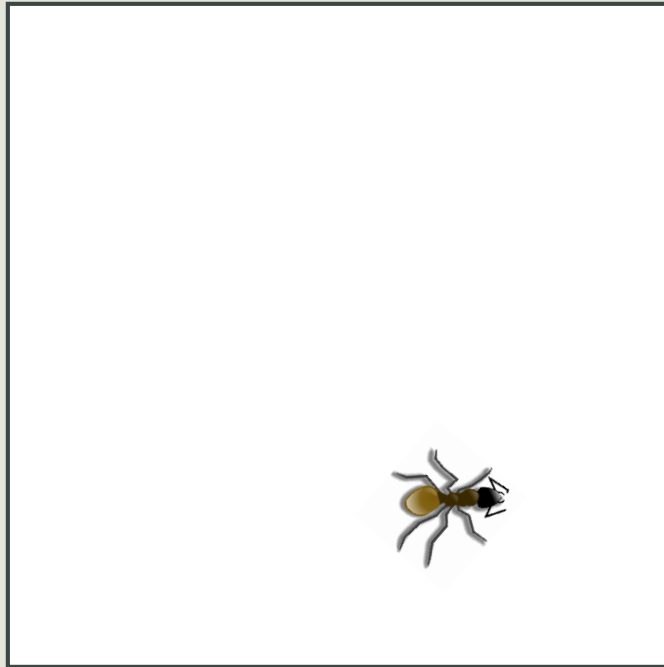
'fff'



Metody – Octave + mrowkaGo

- **f** – idź do przodu jeden krok

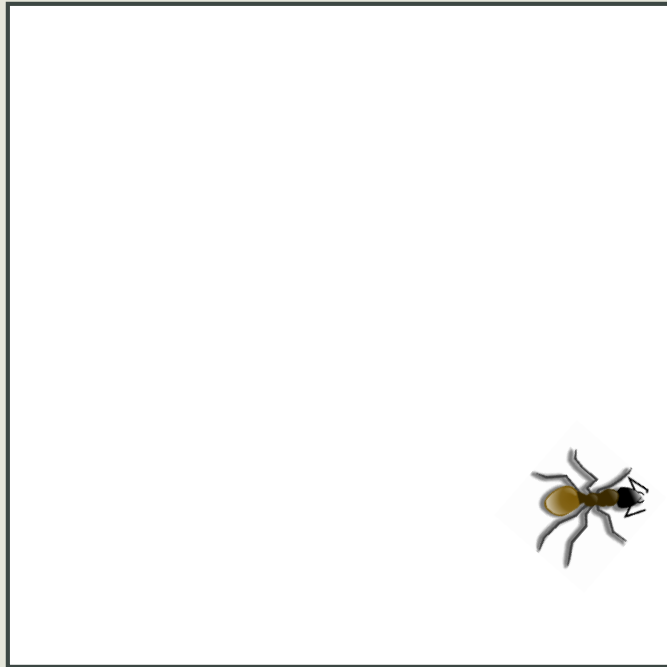
'fff'



Metody – Octave + mrowkaGo

- **f** – idź do przodu jeden krok

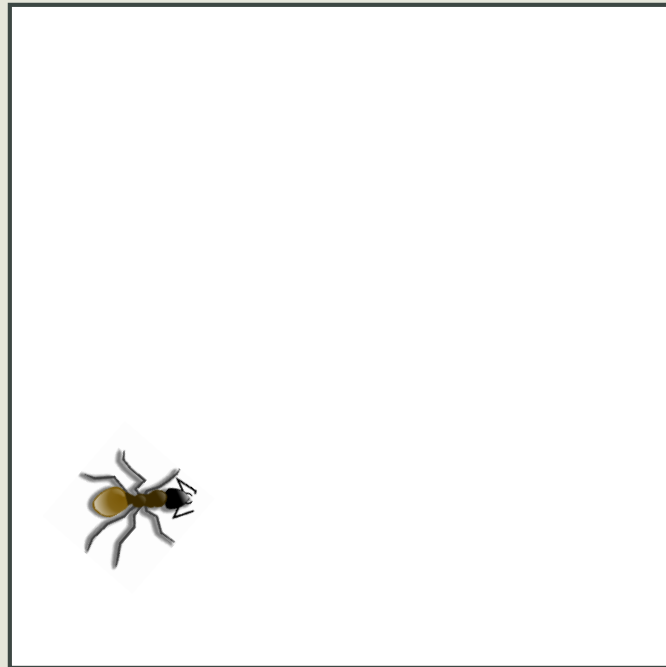
'fff'



Metody – Octave + mrowkaGo

- **F** – idź do przodu jeden krok zostawiając ślad z feromonu

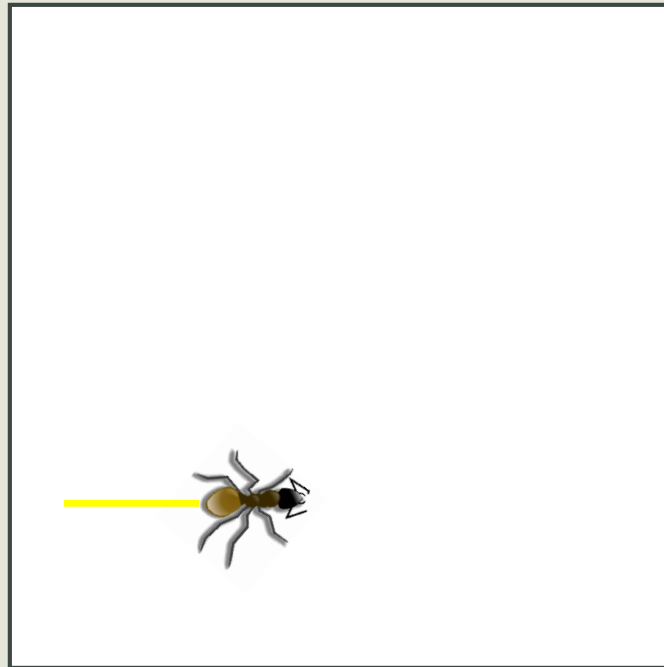
'FfF'



Metody – Octave + mrowkaGo

- **F** – idź do przodu jeden krok zostawiając ślad z feromonu

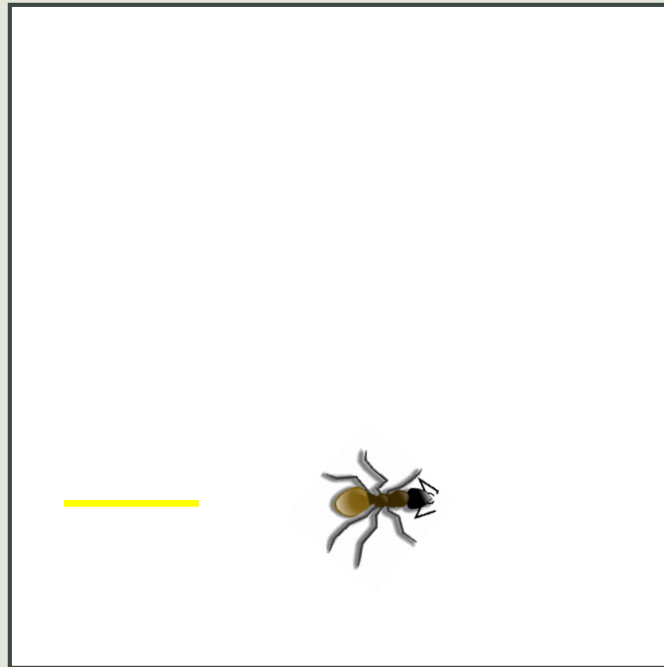
'FfF'



Metody – Octave + mrowkaGo

- **F** – idź do przodu jeden krok zostawiając ślad z feromonu

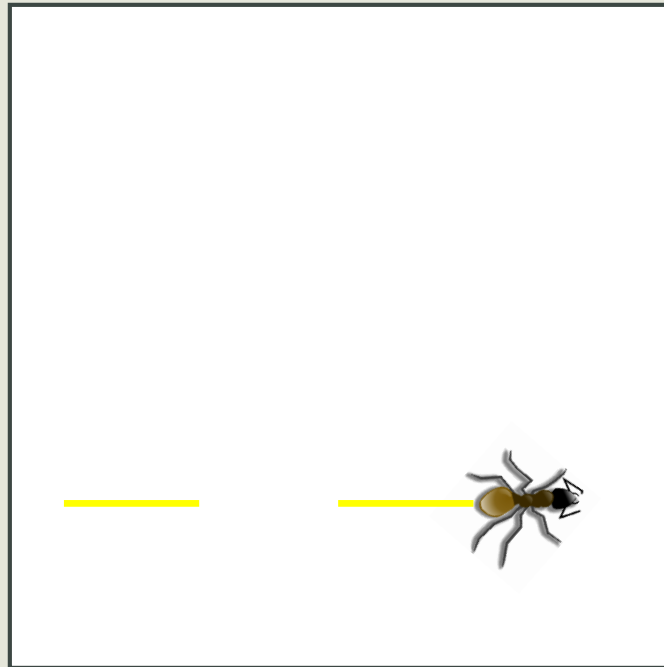
'FfF'



Metody – Octave + mrowkaGo

- **F** – idź do przodu jeden krok zostawiając ślad z feromonu

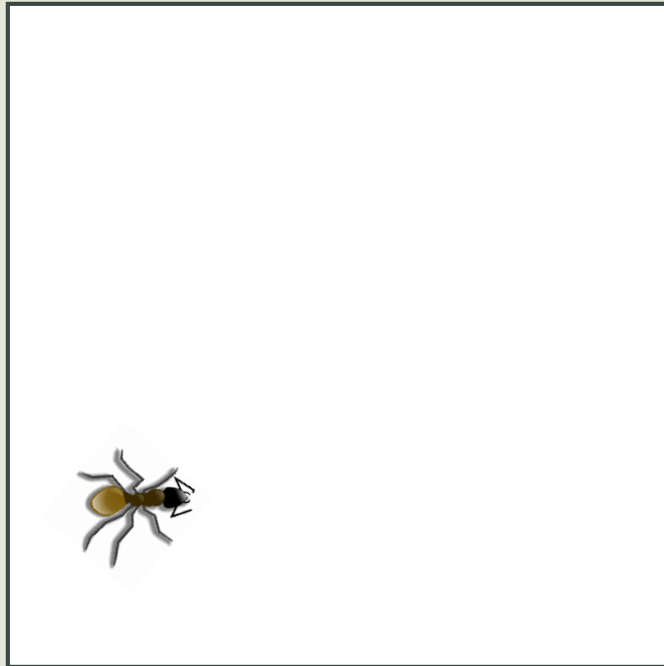
'FfE'



Metody – Octave + mrowkaGo

- **l** – obróć się w lewo (przeciwnie do ruchu wskazówek zegara)
- **r** – obróć się w prawo (zgodnie z ruchem wskazówek zegara)

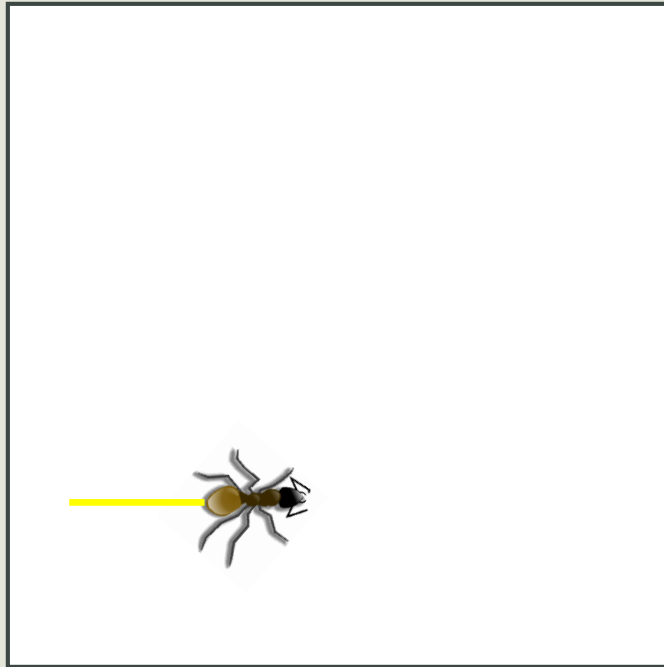
'FlFrF'



Metody – Octave + mrowkaGo

- **l** – obróć się w lewo (przeciwnie do ruchu wskazówek zegara)
- **r** – obróć się w prawo (zgodnie z ruchem wskazówek zegara)

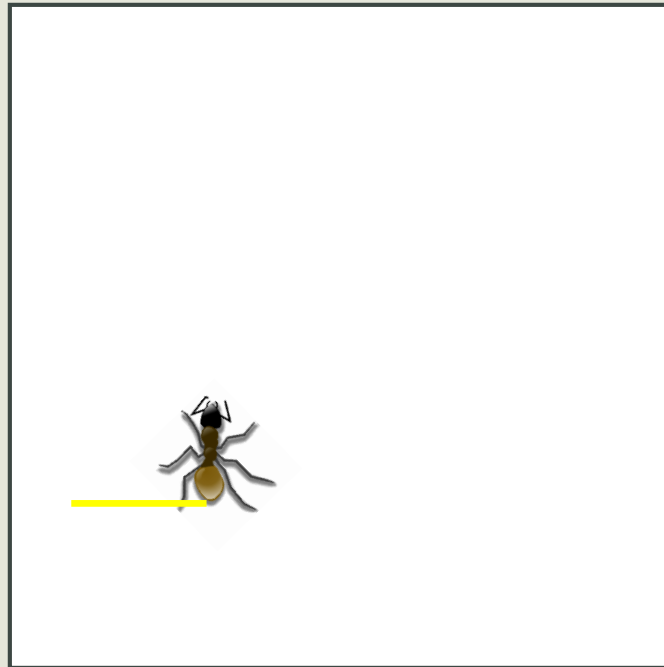
'FlFrF'



Metody – Octave + mrowkaGo

- **l** – obróć się w lewo (przeciwnie do ruchu wskazówek zegara)
- **r** – obróć się w prawo (zgodnie z ruchem wskazówek zegara)

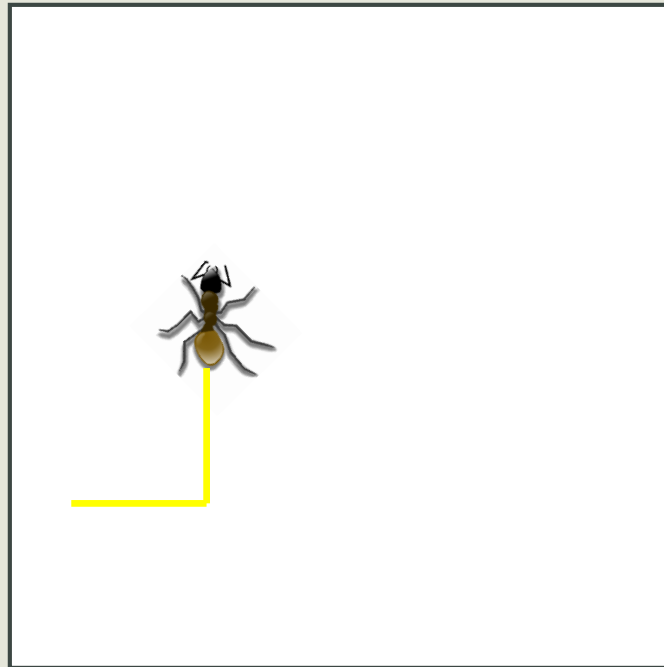
'FlFrF'



Metody – Octave + mrowkaGo

- **l** – obróć się w lewo (przeciwnie do ruchu wskazówek zegara)
- **r** – obróć się w prawo (zgodnie z ruchem wskazówek zegara)

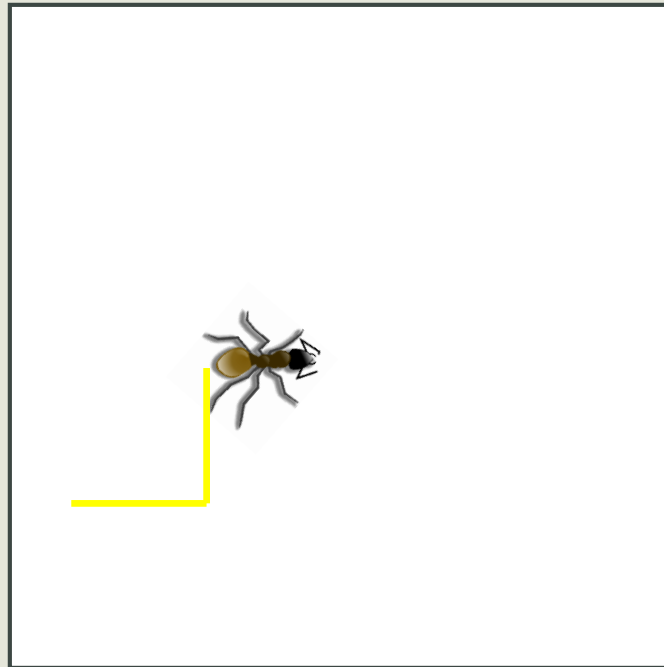
'F**l**ErF'



Metody – Octave + mrowkaGo

- **l** – obróć się w lewo (przeciwnie do ruchu wskazówek zegara)
- **r** – obróć się w prawo (zgodnie z ruchem wskazówek zegara)

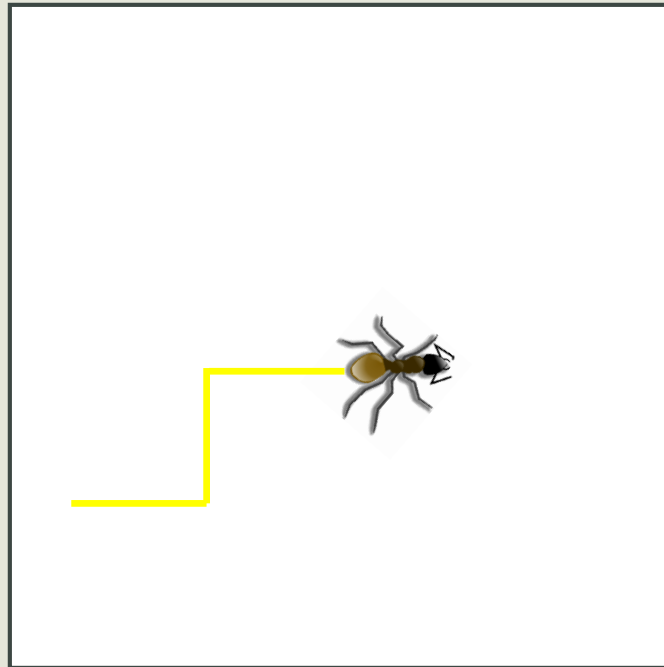
'F**l**FrF'



Metody – Octave + mrowkaGo

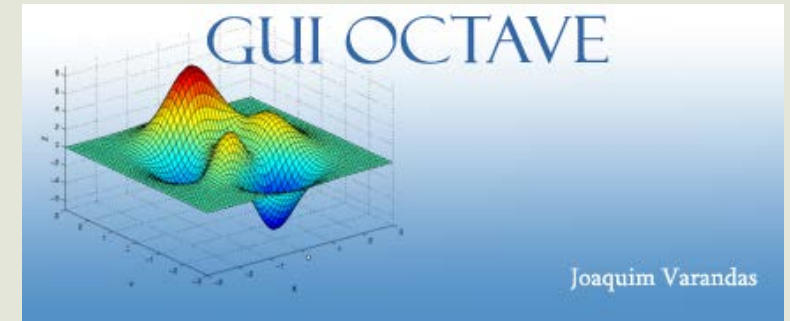
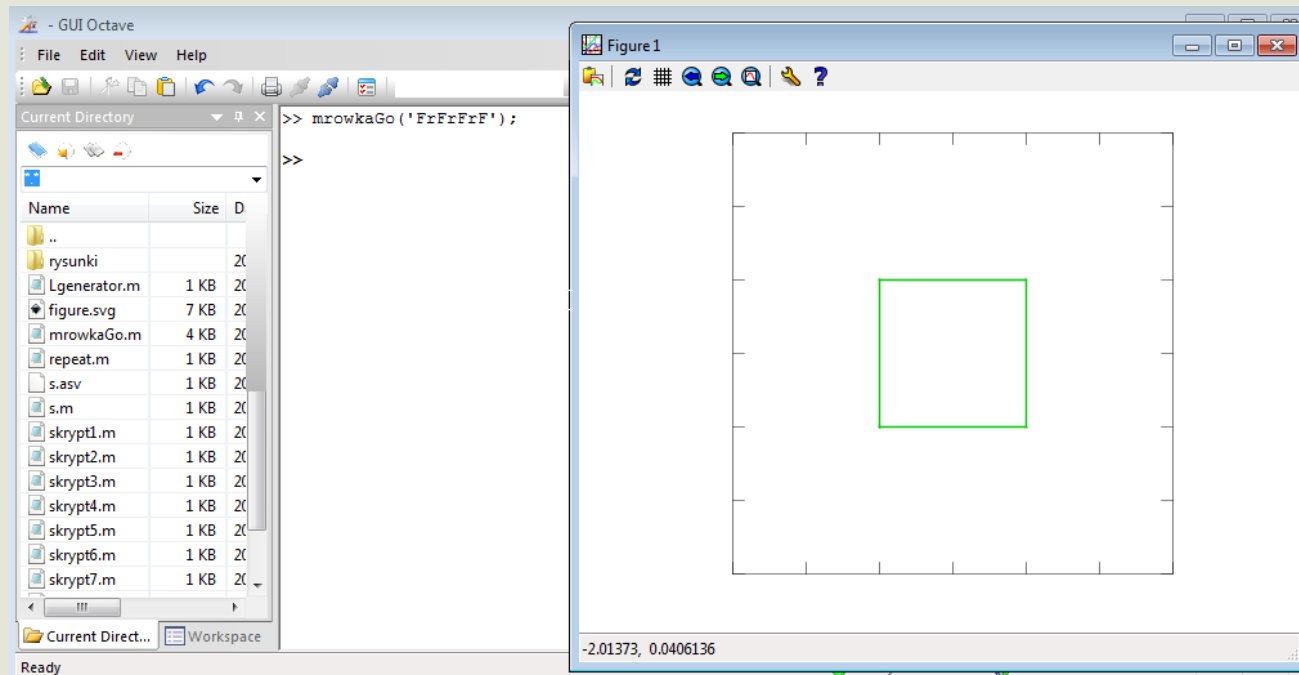
- **l** – obróć się w lewo (przeciwnie do ruchu wskazówek zegara)
- **r** – obróć się w prawo (zgodnie z ruchem wskazówek zegara)

'F $\overline{l}$ FrE'



Metody – Octave + mrowkaGo

- Octave + GUI



`>> mrowkaGo('FlFlFlF')`

Metody – Octave + mrowkaGo

- Paczki

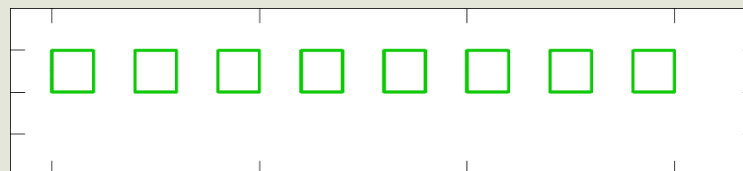
```
kwadrat = 'F\F\F\F'
```

- Powtórzenia I etap

```
repeat(kwadrat, 8)
```

- Powtórzenia II etap – pętla **for**

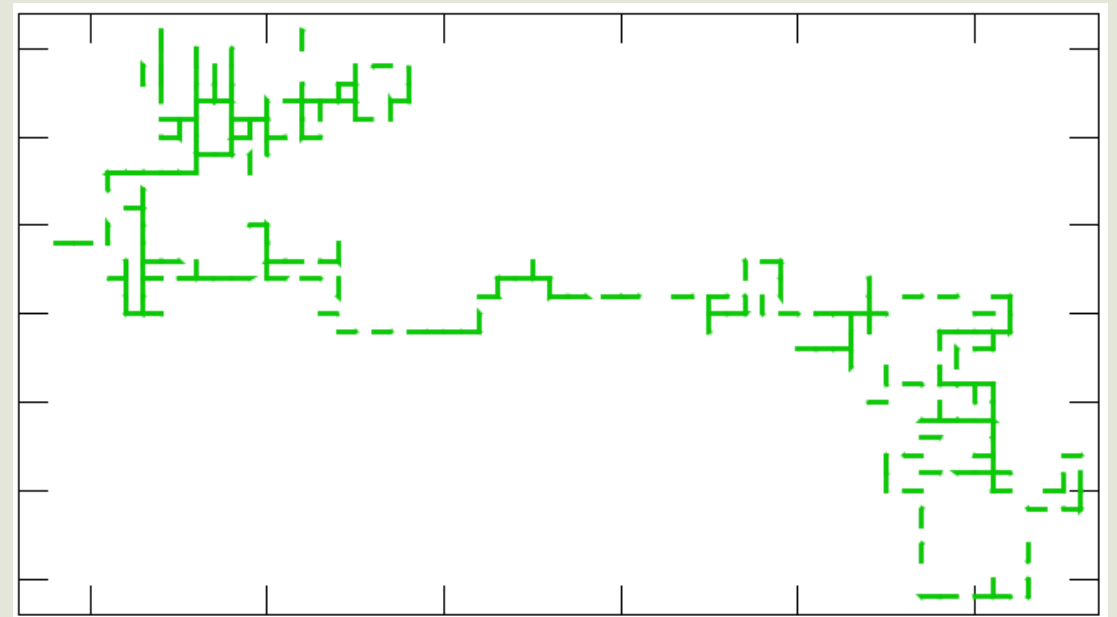
```
paczka = '';  
for licznik = 1:8  
    paczka = [paczka kwadrat];  
end
```



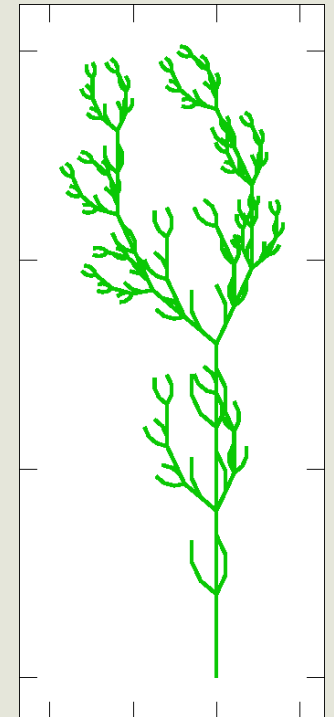
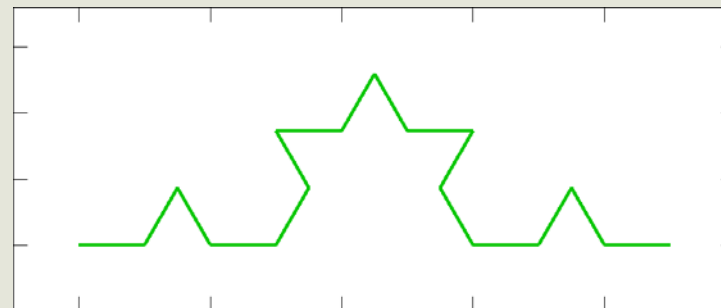
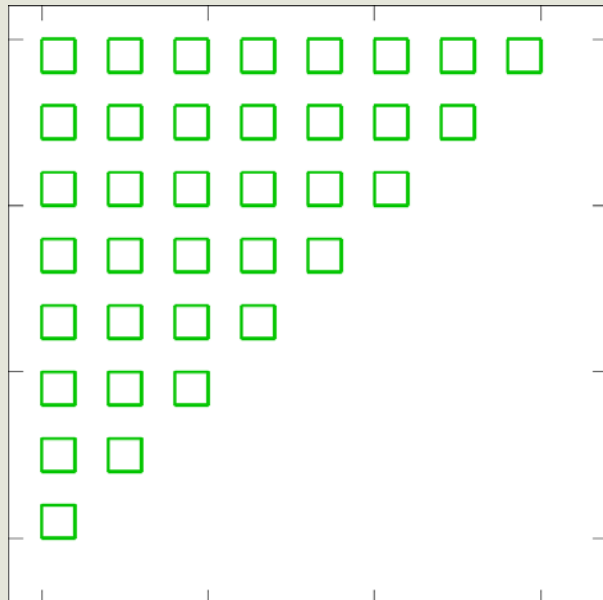
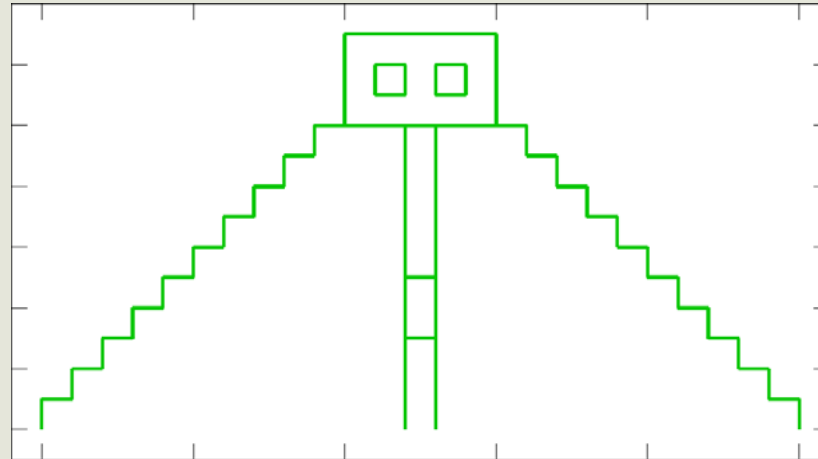
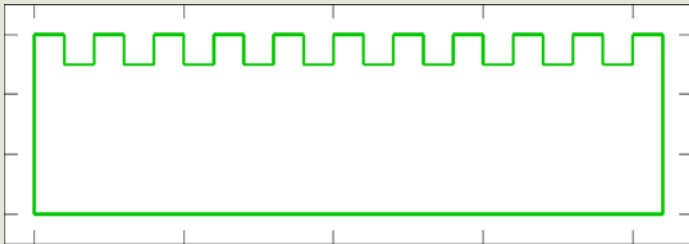
Metody – Octave + mrowkaGo

- Instrukcja warunkowa

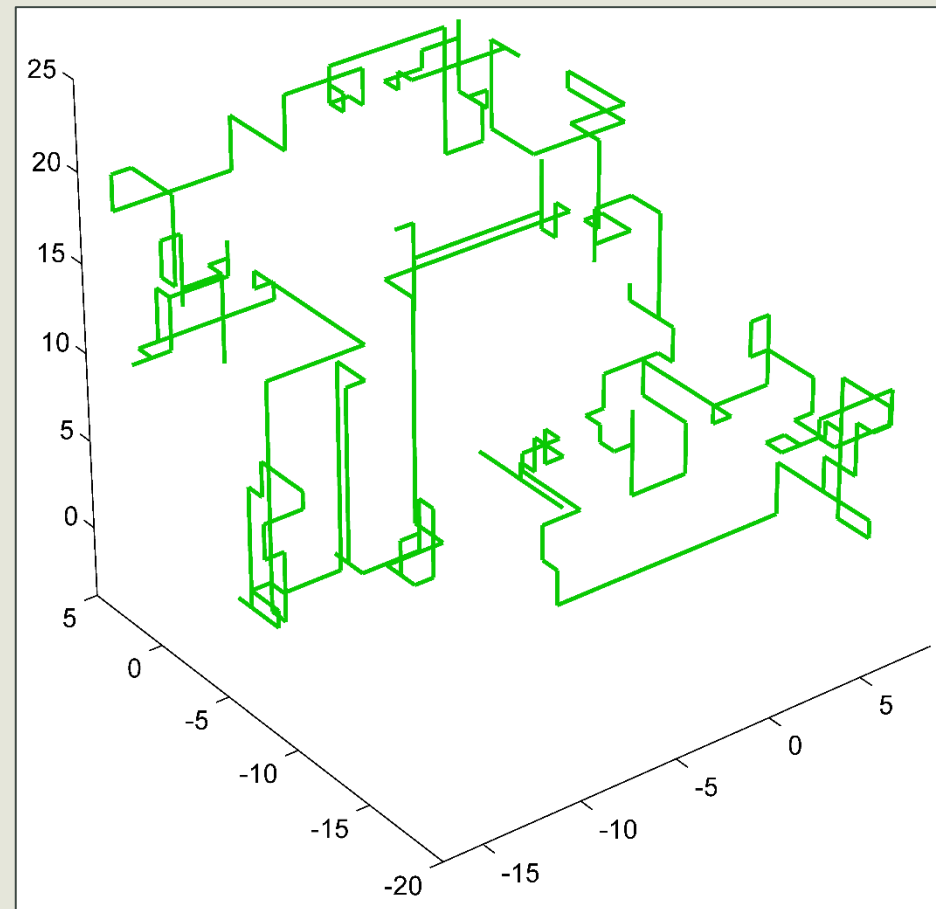
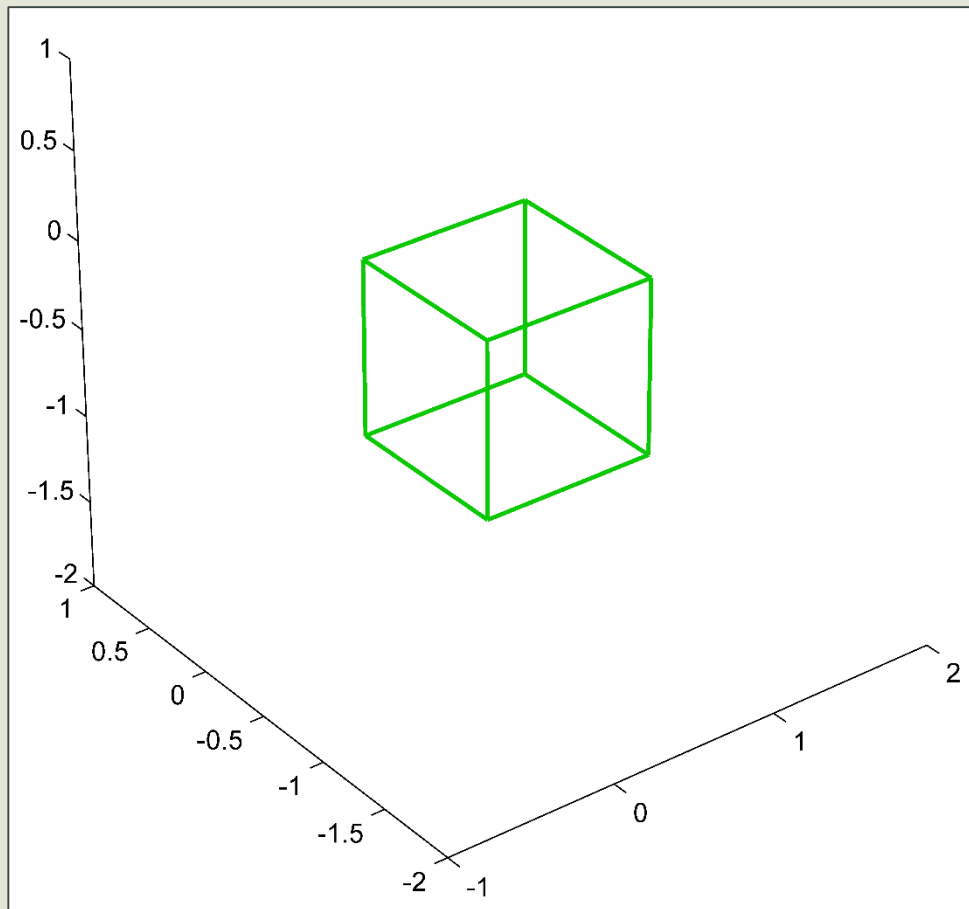
```
spacer = '';
for licznik = 1:1000
    x = randi(4);
    if x == 1
        spacer = strcat(spacer, 'f');
    elseif x == 2
        spacer = strcat(spacer, 'F');
    elseif x == 3
        spacer = strcat(spacer, 'l');
    else
        spacer = strcat(spacer, 'r');
    end
end
```



Metody – Octave + mrowkaGo



Metody – Octave + mrowkaGo



Program nauczania

Etap 1

KoLo

- program
- paczki
- powtórzenia
- warunki

Etap 2

Mrówka

- program
- paczki
- powtórzenia
- warunki

Etap 3

?

- ..
- ..

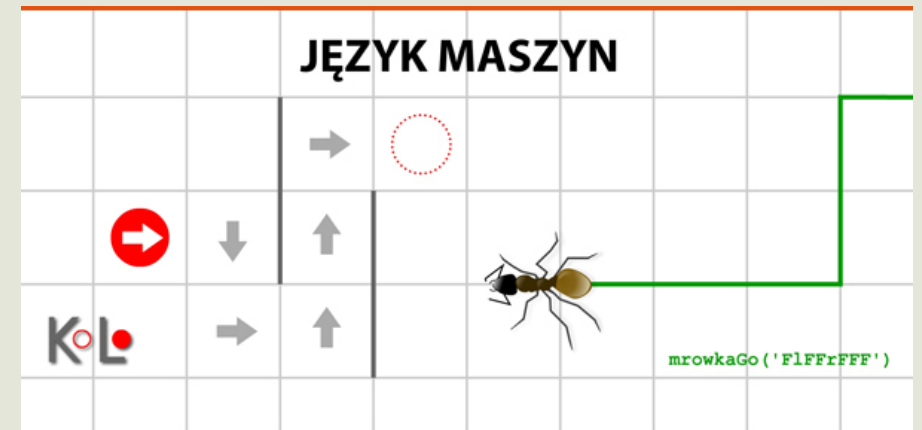
Etap X

Zastosowania

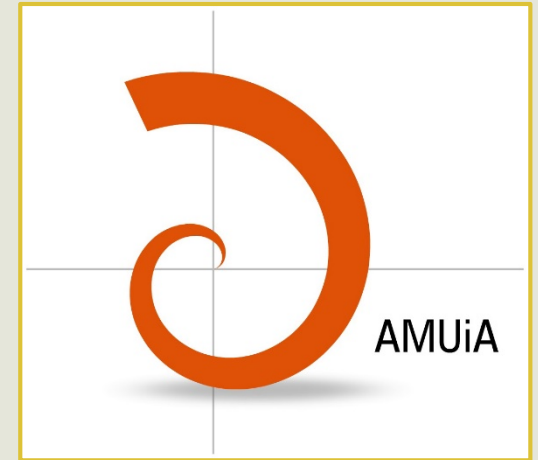
- obraz
- dźwięk
- przyroda
- gospodarowanie
- ...

Pilotaż

- Start: październik 2013
- Jerzy Wachowicz, Wrocławskie Centrum Doskonalenia Nauczycieli – WCDN
- 60 szkół podstawowych i gimnazjalnych z Wrocławia
- Ponad 130 nauczycieli
- Materiały dla nauczycieli: www.jezykmaszyn.pl



Jakub Jernajczyk – *artysta grafik, matematyk*
Bartłomiej Skowron – *filozof, matematyk*
Jarosław Drapała – *informatyk, statystyk*
+
Radosław Rudek – *informatyk*



Język maszyn