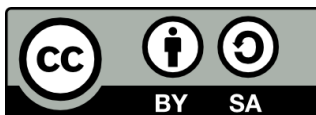


Algorytmika

Czyli jak rozwijać kreatywne myślenie rozwiązując problemy



Damian Kurpiewski

Myśleć jak programista

- Analizować problem, rozpoznawać jego cechy, dzielić na podproblemy
 - Projektować efektywne i uniwersalne rozwiązanie: **algorytm**
 - Znajdować potencjalne luki w swoim rozwiązaniu
 - Myśleć logicznie, szczegółowo i kreatywnie
 - Myśleć **komputacyjnie**
-

Algorytm przejścia przez ulicę

Jeżeli światło **zielone**: idź

Jeżeli światło **czerwone**: stój

Jeżeli światło **zielone**: idź

W **przeciwnym przypadku**: stój



Fragment podstawy programowej



Układa w **logicznym** porządku: obrazki, teksty, polecenia (instrukcje) składające się m.in. na codzienne czynności.



Tworzy polecenie lub **sekwencje poleceń** dla określonego planu działania prowadzące do osiągnięcia celu.



Rozwiązuje zadania, zagadki i łamigłówki prowadzące do odkrywania **algorytmów**.

Założenia



Uczmy **myślenia** i **rozwiązywania problemów**, nie konkretnego narzędzia



Informatyka **to nie tylko komputer** – bez niego też można jej uczyć



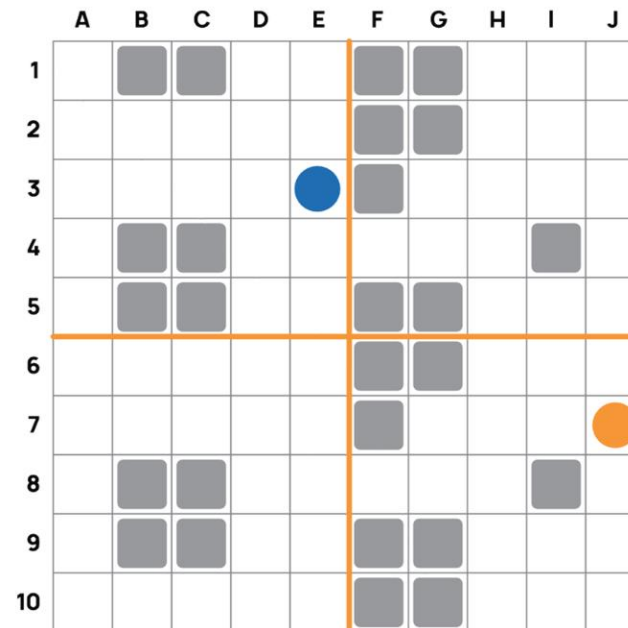
Uczniowie najpierw muszą **zrozumieć temat**, zanim usiądą do komputerów



Celem nauczyciela jest skutecznie **przekazywać wiedzę**, nie ją prezentować

Programowanie bez komputera

- Ćwiczenia związane z rozwiązywaniem problemów
- Tworzymy sekwencję poleceń: dla postaci na kartce, robota, innego ucznia...
- Zaczynamy od prostego rozwiązania (algorytm liniowy), które następnie poprawiamy: stosujemy pętle, instrukcje warunkowe...
- Tworzymy własne scenariusze: zadania dla innych uczniów



- Przykładowa sekwencja dla duszka:
(A – podstawowy, B – z wykorzystaniem pętli)

A  ↓ → → → ↓ ↓ ↓ → → 

B  ↓ 3 → 3 ↓ 2 → 

Zauważcie, że istnieje więcej niż jedno rozwiązanie.

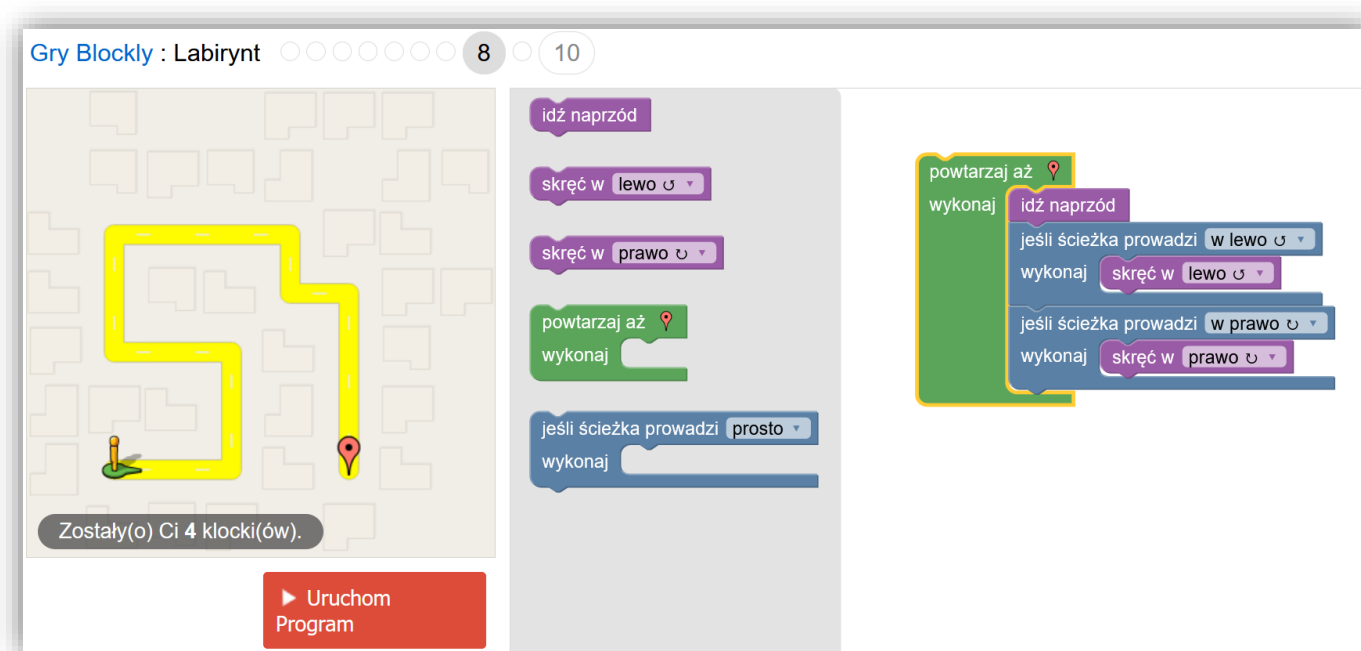
- Alternatywna droga dla duszka:

A  ↓ ↓ ↓ ↓ ↓ → → → ↑ → → 

B  5 ↓ 3 → ↑ 2 → 

Programowanie z komputerem

- **Blockly**: wizualne programowanie
- Nie wymaga obszernego wprowadzenia
- Dostępne są różne ćwiczenia: rysowanie, animacja, muzyka, obliczenia...
- Wymaga jedynie dostępu do Internetu i przeglądarki



Przykładowe zajęcia

1. Zapoznanie się z ideą algorytmiki:
przykłady algorytmów, instrukcje warunkowe, pętle
 2. Ćwiczenia na kartkach:
układamy instrukcje doprowadzające postać do celu, tworzymy własne „mapy” do rozwiązania
 3. Komunikujemy się z komputerem:
Blockly Labirynt, zadania 1-9
 4. Omawiamy i wspólnie rozwiązujemy problem wyjścia z (dowolnego) labiryntu: **algorytm lewej ręki**
 5. **Ewaluacja**: rozwiązujemy zadanie 10 z Blockly Labirynt
-



Źródła

- <https://podstawaprogramowa.pl/Szkola-podstawowa-I-III>
 - <https://www.gov.pl/web/koduj/zabawy-w-programowanie-na-macie--codeweek>
 - <https://blockly.games/maze?lang=pl&level=8>
-