

# STEMKIT

## 4SCHOOLS

### Wprowadzenie do Scratch 2.0

#### PLAN LEKCJI 1



Co-funded by the  
Erasmus+ Programme  
of the European Union

Projekt ten został sfinansowany przy wsparciu Komisji Europejskiej.

Niniejszy komunikat odzwierciedla jedynie poglądy autora, a Komisja nie ponosi odpowiedzialności za jakiegokolwiek wykorzystanie informacji w nim zawartych.



## Spis treści

|       |  |    |
|-------|--|----|
| 1.    | Wprowadzenie do Scratch 2.0.....           | 2  |
| 1.1   | Wprowadzenie.....                          | 2  |
| 1.1.1 | Ogólny opis.....                           | 2  |
| 1.1.2 | Cele nauczania .....                       | 2  |
| 1.1.3 | Odniesienia do programu nauczania.....     | 3  |
| 1.1.4 | Spis niezbędnych materiałów .....          | 4  |
| 1.1.5 | Czas trwania .....                         | 4  |
| 1.2   | Plan lekcji .....                          | 4  |
| 1.2.1 | Wprowadzenie do Scratcha .....             | 5  |
| 1.2.2 | Przygotowanie .....                        | 5  |
| 1.2.3 | Wykonanie .....                            | 7  |
| 1.2.4 | Wnioski .....                              | 11 |
| 1.2.5 | Dodatkowe ćwiczenie (nieobowiązkowe) ..... | 12 |
| 1.3   | Bibliografia i dodatkowe zasoby.....       | 12 |



# 1. Wprowadzenie do Scratch 2.0

## 1.1 Wprowadzenie

### 1.1.1 Ogólny opis

W tym scenariuszu lekcji nauczyciel przedstawi aplikację Scratch, służącą do tworzenia projektów zawierających media i skrypty, oraz język programowania, aby uczniowie mogli odkrywać, wyrażać siebie i uczyć się poprzez doświadczanie. Lekcja zachęca do zgłębiania kluczowych koncepcji i praktyk myślenia obliczeniowego.

Obejmuje to trzy kluczowe wymiary: (1) koncepcje obliczeniowe, (2) praktyki obliczeniowe oraz (3) perspektywy obliczeniowe.

Ucząc się aktywności w społeczności internetowej Scratcha i na warsztatach Scratcha, młodzi ludzie będą tworzyć własne interaktywne historie, gry i symulacje, a następnie dzielić się tymi dziełami w społeczności internetowej z innymi młodymi programistami z całego świata.

Kreatywne przetwarzanie wspiera rozwój osobistych powiązań z komputerami, czerpiąc z kreatywności, wyobraźni i zainteresowań. Studenci będą lepiej przygotowani do przyszłej kariery jako informatyk lub programista.

### 1.1.2 Cele nauczania

Scratch to język programowania, stworzony przez MIT Media Lab, środowisko programistyczne typu open-source, które ułatwia tworzenie interaktywnej sztuki, historii, symulacji i gier. Jego celem jest kształcenie osób z niewielkim lub zerowym doświadczeniem w programowaniu, głównie dzieci w wieku od 8 do 16 lat.

Ta lekcja wprowadza uczniów do podstawowych pojęć związanych z programowaniem komputerowym i umiejętnościami myślenia obliczeniowego, badając aspekty środowiska programistycznego Scratch.

Jest to świetny sposób, aby dzieci mogły zapoznać się z programowaniem bez wcześniejszego doświadczenia w tym temacie. Uczniowie nauczą się importować obrazy i dźwięki stworzone w Scratch, używając interaktywnej sztuki, opowieści, symulacji i gier, narzędzi do malowania budynków i rejestratora dźwięku jako działania eksperymentalnego.



Główne cele dydaktyczne tego planu lekcji to:

- zrozumienie koncepcji i treści programu Scratch 2.0. by zainspirować uczniów do nauki programowania komputerowego podczas pracy nad projektami o osobistym znaczeniu, takimi jak animowane historie i gry.
- zaprojektowanie i przeprowadzenie eksperymentu lub badania naukowego wraz ze zbieraniem danych, analizą i prezentacją wyników, dostarczając narzędzi do rozwiązywania technologicznych wyzwań przyszłości
- zapoznanie się z programem Scratch wykorzystywanym w szkołach w wielu dyscyplinach (matematyka, informatyka, sztuki językowe, nauki społeczne)
- zrozumienie podstawowych struktur programowania, posługiwanie się językiem programowania.

### 1.1.3 Odniesienia do programu nauczania

- ✓ Scratch jest skierowany do młodszych użytkowników niż pozostałe dwa systemy, skupia się na samodzielnym uczeniu się, zawiera narzędzia do rysowania obrazów i nagrywania dźwięków.
- ✓ Scratch opiera się na pomysłach konstruktorów, aby pomóc użytkownikom uczynić ich projekty osobiście angażującymi, motywującymi i znaczącymi.
- ✓ Scratch ułatwia importowanie lub tworzenie wielu rodzajów mediów (obrazów, dźwięków, muzyki); został zaprojektowany, aby zachęcać do tworzenia skryptów, zapewniać natychmiastową informację zwrotną na temat wykonania skryptu oraz uwidaczniać wykonanie i dane.

Uczniowie są w stanie zobaczyć postępy w nauce w sposób wizualny w swoim świecie jako serię różnych fizycznych projektów i konstrukcji.

Domeny, subdomeny, przedmioty/tematy, z którymi można powiązać ten plan lekcji, to:

- Nauki ścisłe (fizyka/chemia/biologia/geologia): metoda naukowa, badania, eksperymenty, analiza i interpretacja wyników
- Informatyka: jednostki przetwarzające i urządzenia peryferyjne, interfejsy, język programowania i główne struktury, kodowanie
- Technologia: elektronika, sprzęt i oprogramowanie open source, czujniki, sygnał cyfrowy, obwody, komputery jednopłytkowe
- Matematyka/statystyka: arkusze kalkulacyjne i podstawowe statystyki



### 1.1.4 Spis niezbędnych materiałów

Do tego planu lekcji (i dla każdej grupy uczniów) - oprócz konsoli STEMKIT - będziemy potrzebowali:

- Sprzętu i urządzenia dla nauczyciela i każdego ucznia
- Komputera PC, laptopa lub tableta z zewnętrzną myszką (większość studentów uważa, że łatwiej jest poruszać się w grze za pomocą myszki zamiast touchpada)
- Słuchawek - są pomocne podczas gry (alternatywnie dźwięk w grze może być ściszony lub wyłączony)
- Internetu - do logowania i gry wieloosobowej wymagany jest dostęp do sieci
- Projektora podłączonego do komputera z otwartym programem Scratch, aby wyświetlić, które bloki i skrypty będą wykonywane, a także fizyczne bloki Scratch (opcjonalnie).

### 1.1.5 Czas trwania

Czas trwania lekcji szacuje się na około 45-60 minut, tj. 1 godzinę lekcyjną.

## 1.2 Plan lekcji

Aktywność studenta będzie koncentrowała się na opanowaniu nowych zaprezentowanych pojęć. Ćwiczenie ze Scratchem i odkrywanie wszystkich możliwości nowopoznanych koncepcji jest niezbędne dla odpowiedniego procesu uczenia się. Ćwiczenia zaproponowane w lekcji mają na celu wzmocnienie podejścia skupionego na uczeniu się poprzez działanie.

Lekcja pomaga uczniom rozwijać i wzmacniać wiedzę i techniki poznane w samouczkach, dotyczące przyszłego miejsca pracy, budowania umiejętności, takich jak współpraca, komunikacja, krytyczne myślenie i myślenie systemowe.

Otwarte środowisko uczenia się daje uczniom swobodę eksperymentowania, zachęcając do kreatywnego wyrażania siebie i rozwiązywania problemów.



## 1.2.1 Wprowadzenie do Scratcha

Zestaw komputerowy zawiera wszystko, co potrzebne, aby wprowadzić uczniów w świat informatyki, elektroniki i kodowania. Zbuduj swój własny, w pełni funkcjonalny komputer i poznaj STEAM.

Wiele dzieci nie ma pojęcia o komponentach swoich urządzeń ani o tym, jak działają podstawowe elementy, takie jak połączenia sieciowe i systemy plików. Ostatecznym celem budowy komputera dla uczniów staje się nauka, jak wszystkie fizyczne części oddziałują na siebie i odnoszą się do tego, co widzą na ekranie.

Po zapoznaniu się z tutorialami, uczniowie odtworzą we własnym środowisku Scratcha działania zaprezentowane podczas lekcji. Poprzez to będą oni zachęceni do odkrywania, począwszy od środowiska przedstawionego w tutorialu, wszystkich możliwości, jakie otwierają się przed nimi dzięki nowo poznany koncepcjom.

Ucniowie zostaną wprowadzeni w koncepcje myślenia obliczeniowego pętli, zdarzeń i równoległości, poznają lepiej koncepcje sekwencji, bloków z kategorii Zdarzenia, Sterowanie, Dźwięk i Wygląd + zbadają różne programy artystyczne w Scratchu, stworzą projekt animowanego teledysku.

Po wykonaniu tego zadania uczniowie zapoznają się z pojęciami *zdarzenie* (jedna rzecz powoduje, że inna rzecz się wydarza) i *równoległość* (rzeczy dzieją się w tym samym czasie). Poprzez odpowiednie ćwiczenia będą w stanie wyjaśnić czym one są i jak działają one w Scratchu.

## 1.2.2 Przygotowanie

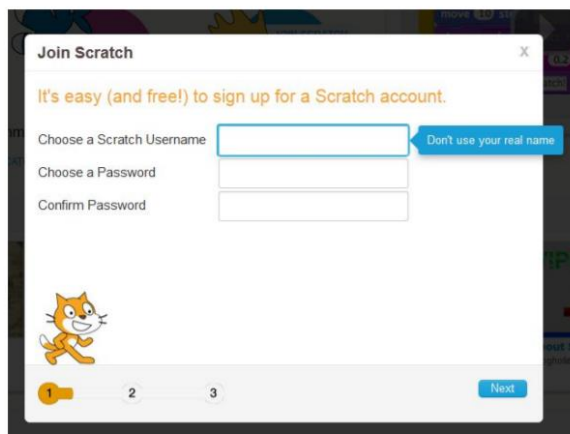
### **Wprowadzenie do budowy programu Scratch**

Głównym motorem zmian była chęć rozszerzenia funkcji edytora i udostępnienia go na szerszej gamie urządzeń. Wersja 2.0 jest oparta na technologii Flash; większość szkół i klubów programistycznych opracowała swoje plany lekcji kodowania Scratch w oparciu o wersję 2.0.

Ucniowie będą tworzyć i odtwarzać projekty na tablecie, laptopie i komputerze stacjonarnym. Będzie dostępny również na smartfonach – choć ich mały rozmiar ekranu może być wyzwaniem. Zaczną odkrywać tę twórczą różnorodność od głębokiego zanurzenia się w animację, sztukę i muzykę.

Istnieją dwie wersje programu Scratch: Scratch 1.4 i Scratch 2.0. W tym kursie będzie wykorzystywany edytor Scratch 2.0 offline. Ale najpierw sprawdźmy jego wersję online. Zalecamy korzystanie z przeglądarki FireFox lub Chrome podczas pracy z tym

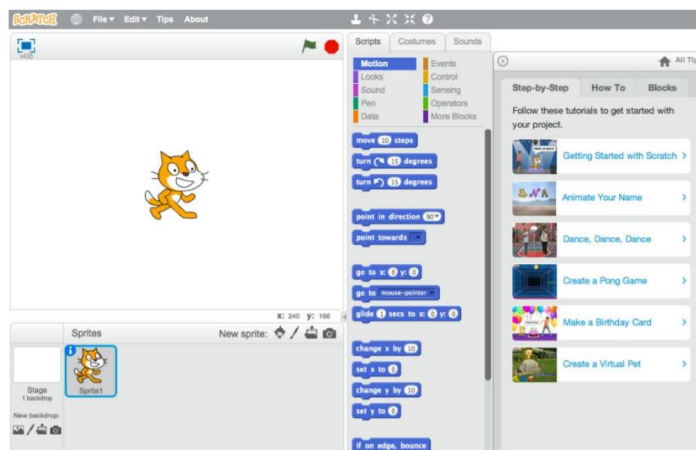
programem. Internet Explorer może sprawiać problemy podczas logowania. Wejdź na stronę Scratch 2.0 dostępną pod adresem [scratch.mit.edu](https://scratch.mit.edu).



Obraz 1. Przejdź do Scratch 2.0 na [scratch.mit.edu](https://scratch.mit.edu)

Kliknij „Utwórz” (create), aby przejść do środowiska programowania Scratch. Oto jak uzyskać dostęp do środowiska programowania online Scratch 2.0. Poświęć trochę czasu na zapoznanie się z witryną [scratch.mit.edu](https://scratch.mit.edu) (kliknij opcje menu Scratch, Create, Explore, Discuss and Help)

Edytor projektu Scratch jest opisany w sekcji Tips, Getting Started, Map of Project Editor. Kliknij, aby wyświetlić szczegóły Edytora projektu. Następnie rozpocznij opcję Krok po kroku, Pierwsze kroki ze Scratchem.



Obraz 2. Nawiguj wprowadzenie krok po kroku

Przejdź przez każdy z 13 kroków we wprowadzeniu Krok po kroku. Przewodnik ten wprowadza podstawowe pojęcia programistyczne i pozwala uczestnikom zapoznać się ze środowiskiem programistycznym Scratch.



Erasmus+

2019-1-FR01-KA201-062281



STEMKIT  
4SCHOOLS



Obraz 3. Kliknij przycisk POMOC na górze, aby rozpocząć. (Źródło: <https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf>)

### 1.2.3 Wykonanie

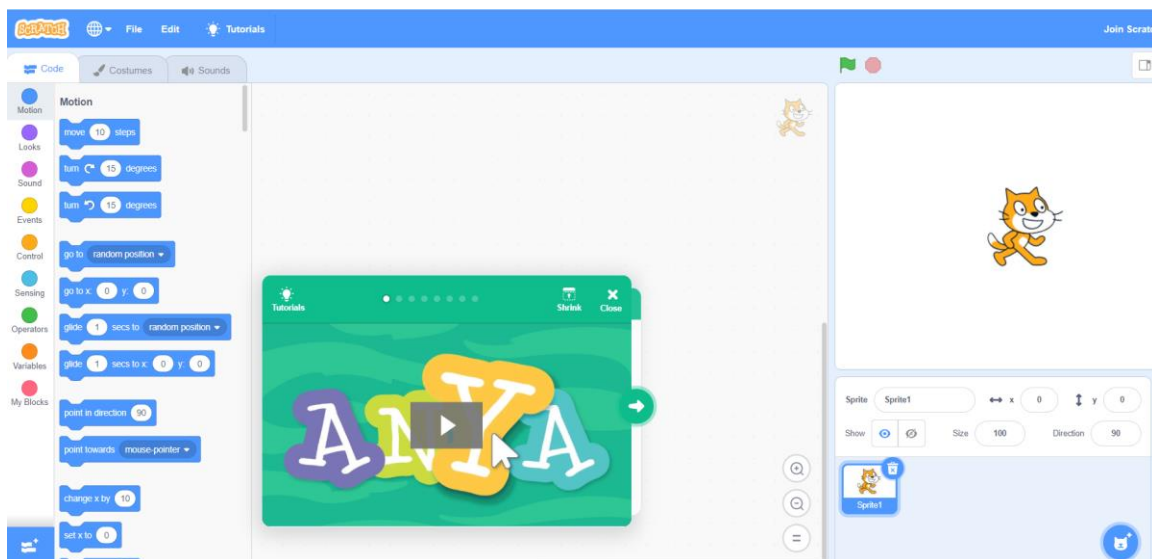
Aktywność Scratch: Animuj swoje imię

Ta aktywność ma pomóc uczniom w odkrywaniu koncepcji obliczeniowych pętli, zdarzeń i równoległości, których kulminacją jest projektowanie spersonalizowanych teledysków. Uczestnicy zdobędą doświadczenie w kodowaniu, animując litery w swoim imieniu.

Najpierw zberz grupę, aby przedstawić temat i zainicjować pomysły. Poproś każdego uczestnika, aby wypowiedział swoje imię, a następnie niech wszyscy w grupie odegrają kształt pierwszej jego litery.

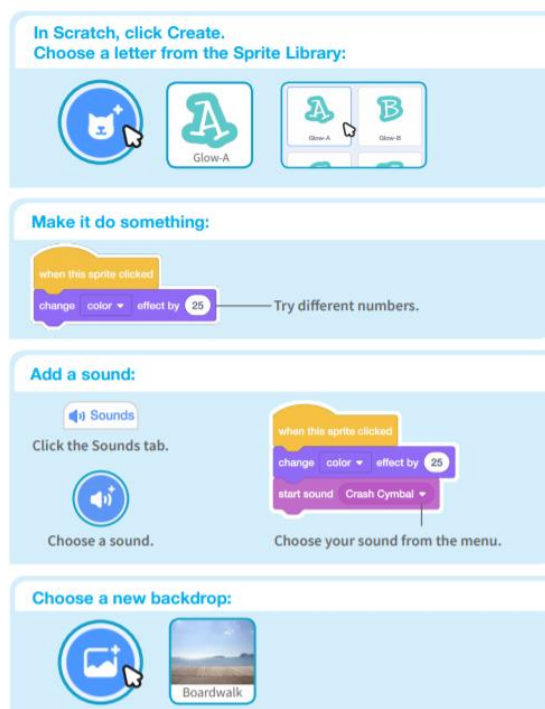
Obejrzyj samouczek: [scratch.mit.edu/name](https://scratch.mit.edu/name) lub [vimeo.com/llk/name](https://vimeo.com/llk/name).



Obraz 4. Tutorial: [scratch.mit.edu/name](https://scratch.mit.edu/name)

Nauczyciel prezentuje film wprowadzający do samouczka *Animuj swoje imię*. Film pokazuje różne projekty, które mogą stać się inspiracją.

Nauczyciel zademonstruje kilka pierwszych kroków samouczka, aby uczniowie mogli zobaczyć, jak zacząć.



Obraz 5. Kroki samouczka: [scratch.mit.edu/name](https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf) (Źródło: <https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf>)



Nauczyciel wydrukuj Karty ćwiczeń - zestawy kart *Animuj swoje imię*, które będą dostępne dla uczestników podczas warsztatów. ([scratch.mit.edu/ideas](https://scratch.mit.edu/ideas)).

Nauczyciele mogą użyć projektora, aby pokazać przykłady i zademonstrować, jak zacząć.

Samouczek *Animuj swoje imię* pokazuje uczestnikom, jak tworzyć własne projekty. Nauczyciel będzie wspierał uczestników podczas tworzenia interaktywnych projektów imion.



**Provide Resources**  
Offer options for getting started



Some participants may want to follow the online tutorial:  
[scratch.mit.edu/name](https://scratch.mit.edu/name)

Others may want to explore using the activity cards:  
[scratch.mit.edu/ideas](https://scratch.mit.edu/ideas)

**Suggest Ideas for Starting**

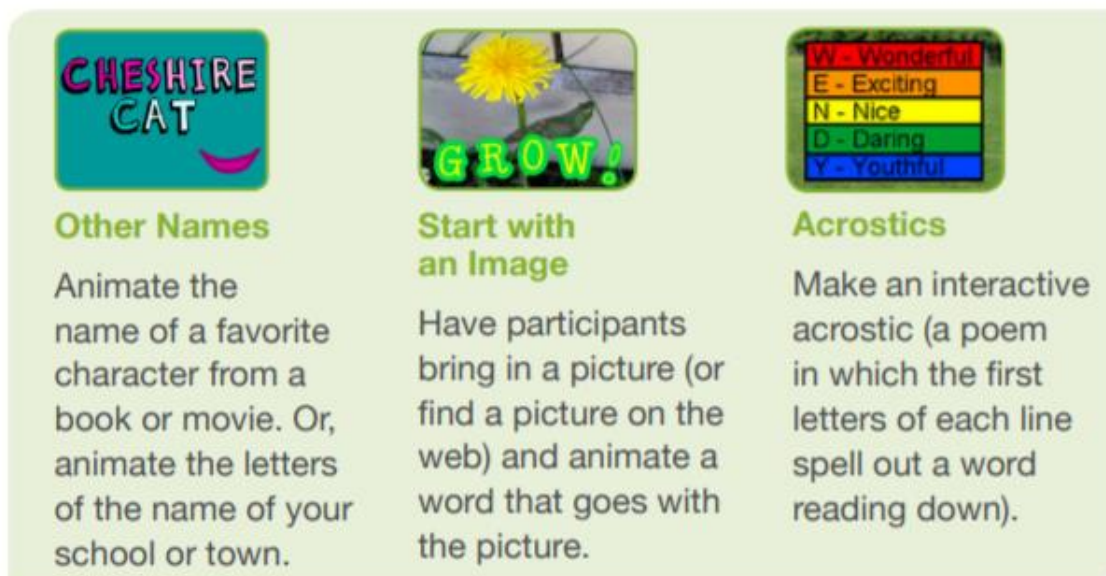
- Choose a letter
- Make it change color
- Add a sound
- Add a backdrop

Obraz 6. Twórz interaktywne projekty imion (Źródło:  
<https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf>)

Gdy ktoś nie będzie w stanie poradzić sobie z wykonaniem zadania, nauczyciel połączy go z innym uczestnikiem, który będzie w stanie mu pomóc.

Nauczyciel pomoże uczniom poczuć się komfortowo, próbując różnych kombinacji bloków aby zobaczyć, co się stanie.

Uczniowie mogą wykorzystać pomysły i koncepcje z tego warsztatu do tworzenia różnorodnych projektów.



Obraz 7. Projekt animowanego imienia (Źródło: <https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf>)

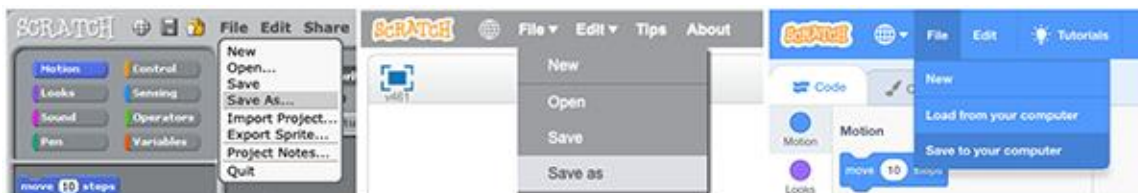
#### 1.2.4 Wnioski

**Scratch** jest jednym z najczęściej używanych narzędzi do kodowania w szkołach.

Środowisko programistyczne i język Scratch współpracują ze sobą, tworząc system, który jest wyjątkowo szybki do nauczenia. Nauczyciele integrują Scratcha z wieloma różnymi przedmiotami i grupami wiekowymi.

Ćwiczenie jest zaprojektowane tak, aby wspierać znajomość i rosnącą biegłość w kreatywności i myśleniu obliczeniowym. Uczestnicy są teraz gotowi do wykonania pracy projektowej z wykorzystaniem zasobów znajdujących się w bazie.

Uczniowie powinni zapisać swoją własną kopię w odpowiednim miejscu w sieci szkolnej z odpowiednią nazwą pliku, np. dodając swoje inicjały do nazwy pliku. Po udostępnieniu projektu, inny użytkownik może dodać komentarze.



Obraz 8. Zapisywanie kopii pliku Scratcha odpowiednio w Scratchu 1.4, 2.0 i 3.0 (Źródło: <https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf>)

W fazie końcowej podsumowujemy to, co zrobiliśmy i w jaki sposób tego dokonaliśmy: jakie były główne etapy, jakie były trudności.

**Podsumowanie:** czego się nauczyłem

Trzy standardowe pytania służące refleksji po zajęciach, aby ocenić wiedzę zdobytą przez uczniów:

- Czego nauczyłeś się dzisiaj, czego nie wiedziałeś wcześniej?
- Co było największym wyzwaniem?
- Jakie masz pytania?

### 1.2.5 Dodatkowe ćwiczenie (nieobowiązkowe)

W ramach kontynuacji tego planu lekcji możemy przejść do następującego ćwiczenia:

*Ćwiczenie Scratch: Wykonaj czynności opisane w samouczku. Eksperymentuj z wielkością i kierunkiem kroków i obrotów.*

## 1.3 Bibliografia i dodatkowe zasoby

Lista przydatnych źródeł i dodatkowych zasobów.

Oto kilka przydatnych źródeł i dodatkowych zasobów związanych z prezentowanym planem lekcji.

- KAY, A. 2010. Squeak etoys, children, and learning. <http://www.squeakland.org/resources/articles>
- Resnick, M., Maloney, j., Monroy-Hernandez, 2009. Scratch: Programming for all. Comm. ACM 52, 11, 60–67.
- ComputerProgrammingInTheEnglishClassroom.pdf



- Maloney, J., Resnick, M., Rusk, N., Silverman, B., and Eastmond, E. 2010. *The scratch programming language and environment*. ACM Trans. Comput. Educ. 10, 4, Article 16 (November 2010), 15 pages. DOI = 10.1145/1868358.1868363. <http://doi.acm.org/10.1145/1868358.1868363>
- <https://education.abc.net.au/home#!/media/1214681/intro-to-scratch-20>
- <https://scratch.mit.edu>
- <http://web.media.mit.edu/~jmaloney/papers/ScratchLangAndEnvironment.pdf>
- <https://www.thomasbuxton.towerhamlets.sch.uk/blogs/year3/2017/11/17/year-3-computing-scratch-projects/>
- <http://scratched.gse.harvard.edu/guide/>
- [scratch.mit.edu/name](http://scratch.mit.edu/name)
- [ScratchManualTermTime20152016.pdf](#)
- <https://resources.scratch.mit.edu/www/guides/en/EducatorGuidesAll.pdf>

#### Słowniczek kluczowych terminów

- Algorytm: zestaw kroków do wykonania zadania w celu rozwiązania problemu.
- Kod lub program komputerowy: zestaw instrukcji, które może wykonać komputer. Na przykład aplikacja lub gra, taka jak Minecraft, to program komputerowy. Terminy mogą być używane zamiennie.
- Sekwencja: identyfikacja serii kroków dla danego zadania
- Pętle: wielokrotne uruchamianie tej samej sekwencji
- Równoległość: sprawianie, że rzeczy dzieją się w tym samym czasie
- Zdarzenia: jedna rzecz powodująca inną rzecz
- Warunkowe: podejmowanie decyzji na podstawie warunków
- Operatory: obsługa wyrażeń matematycznych i logicznych
- Dane: przechowywanie, pobieranie i aktualizowanie wartości