

```
<title>
```

**PIERWSZE
KROKI DZIECKA
W PROGRAMOWANIU**

```
</title>
```

Kids Code Fun
Kidscodfun.com



SPIS TREŚCI

Czym jest programowanie i po co się go uczyć?

Kim jest programista/programistka?

Ale ja nie jestem programistą! Jak odpowiem na pytanie dziecka?

Jak uczy my dzieci?

Jak wprowadzić dziecko w świat programowania?

Narzędzia i scenariusze

CZYM JEST PROGRAMOWANIE I PO CO SIĘ GO UCZYĆ?

Na świecie jest więcej komputerów i telefonów niż ludzi. Jak się z nimi porozumiewamy? Za pomocą języków programowania. Programowanie to rozmowa z maszyną, tworzenie dla niej instrukcji. Jak przepis kulinarny, którego efektem jest konkretnie danie.

Nie każdy musi zostać programistą/programistką. Ale każdemu przyda się umiejętność logicznego myślenia, kreatywność, odwaga do eksperymentowania i popełniania błędów. Analizowania i rozbijania dużych problemów na małe, które można krok po kroku rozwiązać. Tego wszystkiego uczy programowanie.

KIM JEST PROGRAMISTA/PROGRAMISTKA?

To osoba cierpliwa, odważna i kreatywna. Rozwiązuje zagadki, nie bojąc się popełniać błędów. Dzieli się swoją wiedzą z innymi i współpracuje z nimi. To twórca, który potrafi za pomocą zaklęć w języku programowania wyczarować gry lub programy i wpleść trochę magii w otaczające nas przedmioty.

ALE JA NIE JESTEM PROGRAMISTĄ! JAK ODPOWIEM NA PYTANIE DZIECKA?

Wiedzy na świecie jest zbyt wiele, by jakikolwiek nauczyciel znał odpowiedź na każde pytanie. A w programowaniu rzadko kiedy jest tylko jedno właściwe rozwiązanie problemu.

Współczesne narzędzia do uczenia dzieci programowania przypominają klocki i puzzle, które łączymy ze sobą. Dziecko uczy się eksperymentując, doświadczając, mając satysfakcję z samodzielnego odkrywania. Dlatego najważniejszym zadaniem nauczyciela programowania jest zachęcanie dziecka do wspólnego poszukiwania wiedzy, zamiast dostarczania gotowych odpowiedzi.

Najskuteczniejszą metodą uczenia się jest też... uczenie innych. Jeśli czegoś nie wiesz, to doskonała okazja, by pozwolić dziecku wytłumaczyć ci jakie rozwiązania odkryło.

“Ale super, mamy wyzwanie! Spróbujemy to stworzyć razem?”

“To dobre pytanie! Masz pomysł jak to zrobić? Może wypróbuj inne klocki?”

“Jestem pod wrażeniem, wytłumaczysz mi jak to działa?”

JAK UCZYMY DZIECI?

Odwaga do popełniania błędów - chwalimy za próby i zachęcamy do kolejnych

“Jesteśmy chyba coraz bliżej rozwiązania. Spróbujemy jeszcze raz!”

“Jak to śmiesznie zadziałało! Chcemy, żeby tak zostało, czy szukamy dalej?”

Kreatywność - chwalimy nowe pomysły, pozwalamy działać “poza scenariuszem”

“To fajny pomysł. Spróbujesz tak zrobić?”

“Brzmi jak wspaniałe wyzwanie, spróbujemy razem to stworzyć?”

“Nie udało nam się, ale jeszcze się kiedyś nauczymy. Masz pomysł na coś prostszego?”

Poszukiwanie wiedzy - zachęcamy do eksperymentowania

“Spróbujmy razem poszukać odpowiednich bloczków, zobaczymy jak zadziałają.”

“To może nam się kiedyś przydać, przetestujemy jak działa?”

“Tego się nie spodziewałam! Dobrze, że spróbowaliśmy. Eksperymentujemy dalej?”

Dzielenie się wiedzą - zachęcamy dzieci do proszenia o pomoc innych dzieci lub wytłumaczenia czegoś

“Wytłumaczysz mi jak to zrobiłaś? Potrafiłabyś tak zrobić jeszcze raz?”

“Pokażesz koleżance/bratu jak to działa?”

“Sprawdźmy jak to zrobili inni!”

JAK WPROWADZIĆ W ŚWIAT PROGRAMOWANIA?

Bez względu na wiek, warto pierwszy krok w nauce programowania postawić z bezpłatną aplikacją Scratch Jr. Uczy algorytmicznego myślenia, rozwija kreatywność i tworzenie w niej to świetna zabawa.

Wprowadzenie

Zapytaj dziecko:

Czy wie, jaki jest najpopularniejszy język na świecie?

Czy wie, jakim językiem rozmawiamy z komputerem, telefonem pralką?

Wytłumacz:

Z komputerami, robotami i urządzeniami rozmawiamy za pomocą języków programowania. Programowanie, to tłumaczenie komputerowi krok po kroku, co chcemy, żeby dla nas zrobił.

1 Zabawa: **Rysowanie z zamkniętymi oczami**

Poproś dziecko, by narysowało prosty rysunek, ale nie pokazywało efektów swojej pracy. Potem weź kartkę, kredkę i zamknij oczy. Poproś dziecko, żeby “zaprogramowało cię” jak rysującego robota i powiedziało krok po kroku, co masz narysować. Może używać tylko prostych instrukcji takich jak: przesun dłoń w prawo, przyciśnij kredkę do kartki, oderwij kredkę itp.

Porównajcie rysunki, czy są podobne? Czy instrukcje były wystarczająco szczegółowe?

2 Zabawa: **Ja-robot**

Do zabawy wystarczysz ty, dziecko i krzesło. Stań w odległości ok. 2 metrów od krzesła, tyłem do niego. Włóż ręce w kieszenie lub ułóż na swoich biodrach.

Powiedz dziecku: Ja jestem robotem, ty programistą. Powiedz mi, co mam zrobić, żeby usiąść na krześle.

Jeśli pracujesz z grupą dzieci, wszystkie mogą w tej zabawie dawać instrukcje, wybierz dowolne jeśli pojawia się więcej niż jedno polecenie.

Wskazówki dla nauczyciela-roboty: Wykonuj polecenia dosłownie, czasami z przesadą. Gdy dziecko mówi, żebyś zrobił krok, zrób trochę za duży lub w niespodziewanym kierunku. Gdy powie, żebyś podszedł do krzesła, powiedz, że nie masz czujników i nie wiesz gdzie jest krzesło. Gdy uda ci się usiąść, pogratuluj stworzenia pierwszego programu.

Powiedz dziecku:

Pracowałaś jak programistka. Mówiłaś mi co mam robić krok po kroku. Wypróbowałaś różne instrukcje, czasami nie wychodziło, ale potem poprawiałaś. Instrukcje musiały być dokładne, nie wystarczyło “zrób krok”, potrzebny był kierunek albo informacja jak duży powinien być krok.

Jeśli pracowałaś z grupą, dodaj: Tworzyliście instrukcję razem i poprawialiście sobie nawzajem te instrukcje, które nie działały lub nie były dokładne.

Właśnie stworzyliście swój program bez pomocy komputera. Pamiętajcie przy programowaniu o tej zabawie i o tym, co było w niej ważne: dokładne instrukcje, poprawianie instrukcji, poprawianie błędów. Robot mógł dojść do krzesła wieloma ścieżkami, niektóre mogą być krótsze, niektóre dłuższe. Ale najważniejsze, gdy program działa, czyli udało się doprowadzić robota we właściwe miejsce.

3 Zabawa: Ty-robot

Do zabawy wydrukuj z zestawu bloczków ScratchJr dostępnych [tutaj](#) bloczki:

- Zielona flaga na żółtym bloczku
- Prosta strzałka w lewo na niebieskim bloczku (Move Left)
- Zaokrąglona strzałka w prawo na niebieskim bloczku (Turn Right)
- Prosta strzałka w prawo na niebieskim bloczku (Move Right)
- Czerwony bloczek bez symbolu (End)

Ułóż bloczki jeden obok drugiego, od lewej strony do prawej, według kolejności powyżej.

Powiedz dziecku:

Teraz ja ułożyłem program z puzzli. Zielona flaga pokazuje start programu, potem jest instrukcja jak masz się poruszać. Jak myślisz, co oznaczają niebieskie bloczki?

[Odp.: ruch w lewo, w prawo - UWAGA! Prosta strzałka w prawo/lewo oznacza przesunięcie się w prawo/lewo, nie obrót].

Powiedz dziecku:

Teraz ty bądź robotem i wykonaj program, który ułożyłem/ułożyłam.

4 Programowanie: Scratch Jr - puzzle i zeszyt

Włączcie aplikację ScratchJr i zainicjujcie w niej nowy projekt.

Powiedz i pokazuj dziecku: Tu po środku jest scena, na której jest kot. Możesz dawać mu instrukcje tak jak ja tobie lub ty mi wcześniej. Biały pasek u dołu to zeszyt, w którym układasz instrukcje. Instrukcje układasz jak puzzle, łącząc je ze sobą. Tak jak z puzzlami, muszą być połączone ze sobą, dotykać się i pasować do siebie. Różne puzzle w różnych kolorach są w szufladkach. Aby otworzyć szufladkę z niebieskimi bloczkami, naciskasz guzik w kształcie niebieskiego bloczku po lewej nad zeszytem. Wtedy obok pojawiają się wszystkie niebieskie bloczki. Spróbuj wyciągnąć jakieś bloczki z szufladek i ułożyć pierwszą instrukcję na dole w zeszytce. Aby uruchomić program zaczynający się od zielonej flagi, trzeba nacisnąć zieloną flagę nad sceną, po której porusza się kot.

Spróbuj stworzyć animację z kilkoma poruszającymi się zwierzątkami!

Ekran programu Scratch Jr



BEZPŁATNE SCENARIUSZE I PROGRAMY

(kliknij aby przejść do strony)

Scenariusze

[Materiały zebrane przez Ministerstwo Cyfryzacji](#)

[Scenariusze Koduj z Klasą](#)

[Scenariusze Link do Przyszłości](#)

[Zabawa w programowanie bez komputera z Hello Ruby](#)

[Super Koderzy](#)

[Mistrzowie Kodowania](#)

[Scenariusze Uniwersytetu Warszawskiego](#)

[Fundacja Chocimska scenariusze zajęć z robotami Lego i Scratchem](#)

[ScratchJr](#) (po angielsku)

[Scratch](#) (po angielsku)

[Pluralsight](#) (po angielsku)

[Kodable](#) (po angielsku)

Programy

[ScratchJr](#) (wiek 5+)

[Daisy the Dinosaur](#) (wiek 5+)

[Lightbot](#) (wiek 5+)

[Hopscotch](#) (wiek 5+)

[Swift Playgrounds](#) (wiek 5+)

[Code.org](#) (wiek 6+)

[Tynker](#) (wiek 7+)

[Scratch](#) (wiek 8+)

[Pocket Code](#) (wiek 8+)

[Khan Academy po polsku](#) (wiek 9+)

[AppInventor](#) (wiek 10+)

[Code Combat](#) (wiek 10+)

Książki, roboty i inne narzędzia

[Inteligentne Miasto](#)

[Roboty Dash&Dot](#)

[Seria książek "Programowanie na wesoło" PWN](#)

[Książka Hello Ruby Sierra Madre](#)

[Lego WeDo](#)

[Lego Mindstorms](#)

[Klocki Cubetto](#)

[Komputer Kano](#)

[Microbit BBC](#)

[Arduino](#)

[Makey makey](#)

Ciekawe strony i blogi

[Kodowanie na dywanie](#)

[Mali programiści](#)

[Oswajamy programowanie](#)

[Geek Tata](#)