

**Temat:** Międzynarodowy Dzień Kropki

**Przedmiot:** zajęcia komputerowe/nauczanie zintegrowane

Klasy: 3-6

**Autor:** Sylwester Zasoński

**Czas trwania:** 1h lekcyjna

**Cele ogólne:**

- Rozwijanie kompetencji miękkich
- Pobudzanie kreatywności
- Łączenie kluczowych obszarów rozwoju

**Cele operacyjne:**

Uczeń:

- Potrafi szukać sposobów rozwiązania zadania czy problemu
- planuje i realizuje własne projekty/prace; realizując te projekty/prace współdziała w grupie
- programuje wizualnie: proste sytuacje lub historyjki według pomysłów własnych i pomysłów opracowanych wspólnie z innymi uczniami, pojedyncze polecenia, a także ich sekwencje sterujące obiektem na ekranie komputera bądź innego urządzenia cyfrowego
- zapisuje efekty swojej pracy we wskazanym miejscu.
- posługuje się komputerem lub innym urządzeniem cyfrowym oraz urządzeniami zewnętrznymi przy wykonywaniu zadania
- korzysta z udostępnionych mu stron i zasobów internetowych
- tworzy polecenie lub sekwencje poleceń dla określonego planu działania prowadzące do osiągnięcia celu
- współpracuje z uczniami, wymienia się z nimi pomysłami i doświadczeniami, wykorzystując technologię

**Metody:**

Poszukujące, eksponujące

**Środki dydaktyczne:**

1. Robot Codey Rocky
2. Komputer z aplikacją mBlock oraz adapterem Bluetooth od Makeblock lub kablem z zestawu do wgrania kodu

**Przebieg zajęć:**

Zajęcia rozpocznij od obejrzenia bajki o dziewczynce Vashti lub przeczytania ilustrowanego opowiadania Peter'a H. Reynolds.

Link do nagrania:

<https://youtu.be/7CRNTbFRJZo>

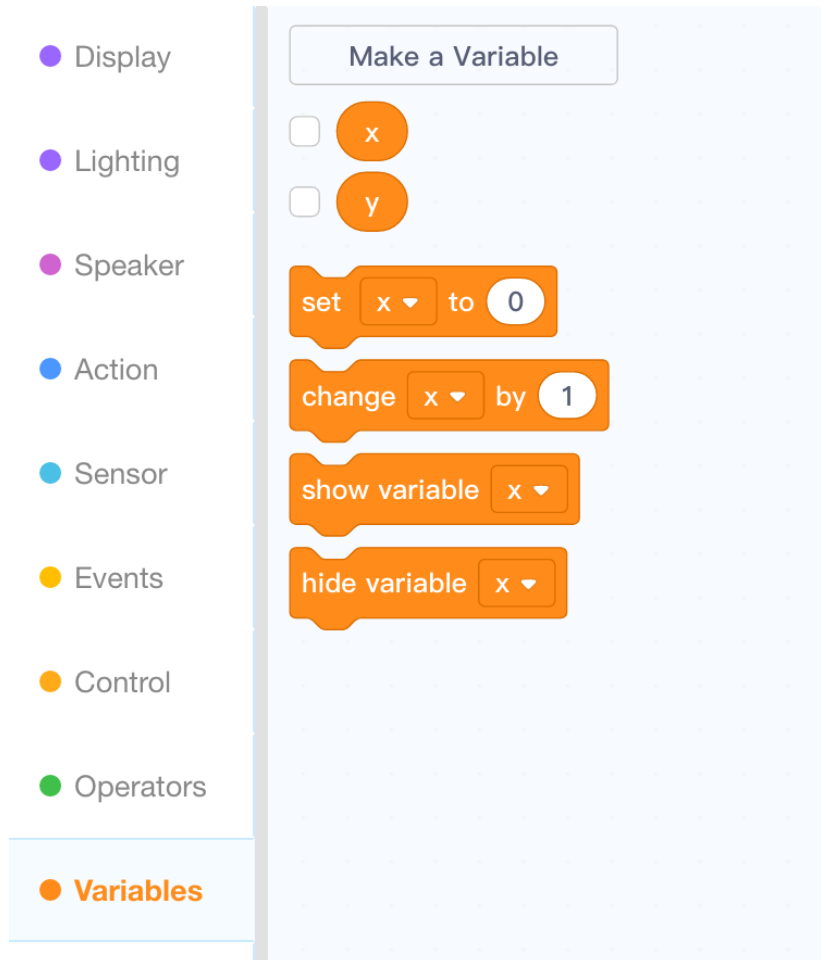
Porozmawiaj z uczniami nt. opowiadania. O tym jak czuła się Vashti i jakie przesłanie niesie ta opowieść.

Wy tłumacz, że wy również będziecie rysować kropki, a dokładniej cyfrowe rysunki złożone z kropek. Ale zamiast kartki papieru i kredek użyjecie robota Codey Rocky (a dokładniej samej głowy, czyli Codey). Sterując przyciskami będziesz poruszać się po wyświetlaczu zapalając wybrane diody led.

Uruchom zainstalowaną na komputerze aplikację mBlock5.

Pierwszy kod to ustawienie pozycji startowej, czyli miejsca, z którego będziesz zaczynać. W tym celu potrzebujesz utworzyć 2 zmienne: x i y.

Zmienna to taki schowek gdzie będziesz przetrzymywać dane.



Wybierz kategorię Variables – zmienne i kliknij *Make a variable* - utwórz zmienną. W tym przypadku nie będzie różnicy czy zmienna będzie dotyczyć wszystkich duszków czy jednego, gdyż wykorzystamy ją do zaprogramowania ekranu Led w Codey'u.

Pierwszy kod ustawiający pozycję początkową powinien wyglądać następująco:

*Kiedy Codey uruchomiony*

*Ustaw x na 0*

*Ustaw y na 0*

Sekwencja ustawia wartości dwóch zmiennych (x i y) na 0.



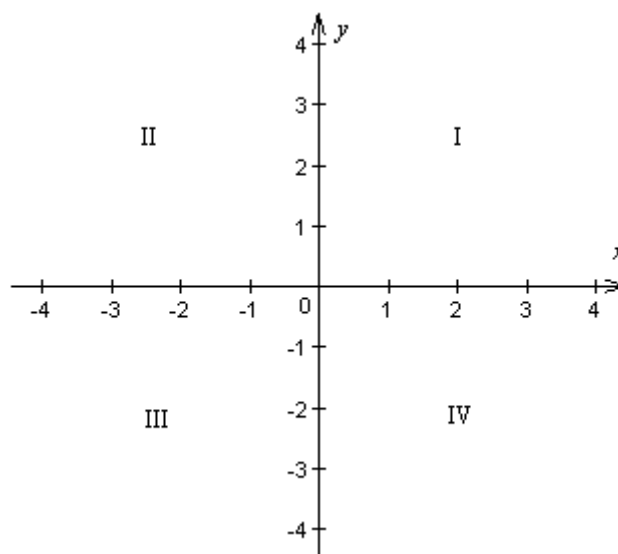
Kolejny punkt to zaprogramowanie przycisków, aby za ich pomocą przesuwać się po ekranie led Codey'a.

Bloczki "Kiedy przycisk A naciśnięty" znajdziesz w kategorii *Events* – wydarzenia.

Przycisków masz 3, więc braknie jednego na jeden kierunek, ale obejdziesz to innym sposobem, o tym poniżej.

Po panelu poruszać się będziesz według kordynatów znanych ze Scratch czyli osi X i osi Y. Przyjmijmy zatem że przyciski A i B będą zmieniać położenie w poziomie czyli na osi X. Do poruszania po osi Y użyjesz przycisku C oraz potrząśnięcia robotem. Tak kiedy Codey potrząśnięty to pozycja Y zmieni się o  $-1$ .

I tak przycisk A będzie zmieniać zmienną X (czyli pozycję w poziomie) o  $-1$ , przycisk B o  $1$ . Przycisk C zmieni natomiast Y (pozycję w pionie) o  $1$ , a potrząśnięcie o  $-1$ .



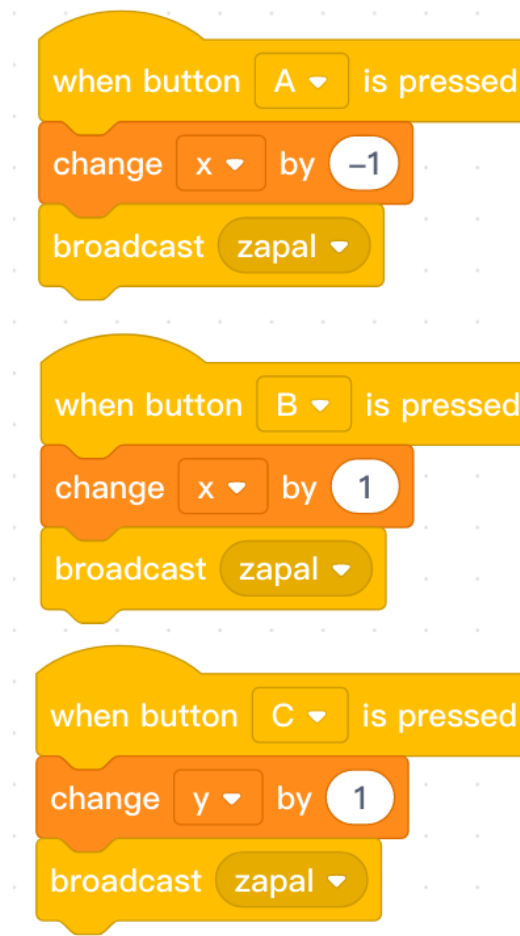
Powyższy obraz układu kartezyjskiego może pomóc w zorientowaniu się w kierunkach poruszania.

Każdy z powyżej opisanych kodów powinien się kończyć wysłaniem wiadomości.

Dlaczego wysłaniem wiadomości? Ponieważ w osobnej sekwencji napiszemy kod zapalający diodę w pozycji w której aktualnie się znajdujemy.

Wiadomości znajdują się również w kategorii *Events* – wydarzenia. W tym przypadku do nadania potrzebny będzie bloczek *Broadcast* – nadaj. Po kliknięciu rozwijanego menu wybierz *New Message* – nowa wiadomość i nadaj jej własną nazwę np. Zapal.

Kod dla przycisków A,B,C powinien wyglądać tak:



Trochę bardziej skomplikowanie wygląda kod z wykorzystaniem potrząśnięcia Codey'a. Użyjesz w nim zarówno pętli *powtórz jak* i warunku *jeżeli*.

Podobnie do początkowego ustalania pozycji tu też kod zostanie uruchomiony wraz ze startem robota. Pierwszy będzie bloczek *Kiedy Codey uruchomiony*. Kluczem będzie bloczek, który znajduje się w kategorii *sensory*, czyli *shaking* - potrząśnięty.

Potrzebujesz taki warunek:

*Kiedy Codey potrząśnięty wtedy*

*Zmień y o -1*

*Nadaj zapal*

*Aby warunek był sprawdzany nieustannie, potrzebna będzie jeszcze pętla zawsze. Cały kod wyglądał by tak:*

*Zawsze*

*Kiedy Codey potrząśnięty wtedy*

*Zmień y o -1*

*Nadaj zapal*

Ułożony kod poniżej:



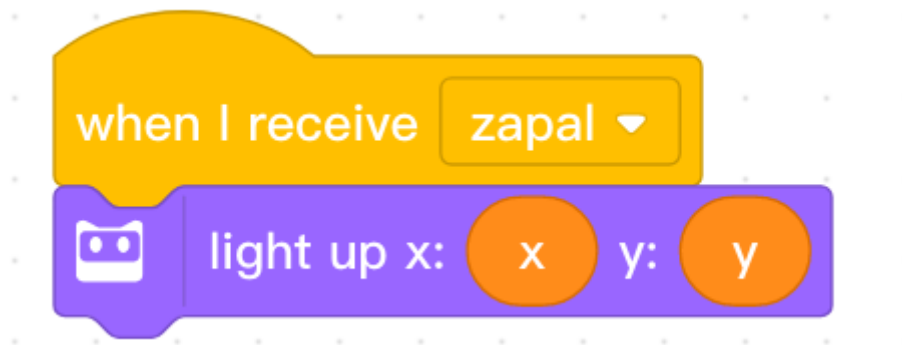
Wszystkie powyższe sekwencje odpowiadają za poruszanie się po ekranie Codey'a. Każdy z nich kończy się nadaniem komunikatu *Zapal*. Potrzebujesz teraz jeszcze sekwencji co się będzie działo, gdy taka komenda zostanie odebrana. Wyszukaj zatem *When I receive*, które znajdziesz w kategorii *Events*.

W najprostszej wersji tej sekwencji, kolejny bloczek będzie odpowiadał za zapalenie diody led w pozycji przechowywanej w dwóch zmiennych x i y.

Potrzebujesz bloczek *light up x:0 y:0*. Znajdziesz go w kategorii *Display* - wygląd.

W miejsce pierwszego zera wstaw zmienną x (kategoria *variables*) a w miejsce drugiego zera (po y) wstaw zmienną y.

Zwróć uwagę na kształt zmiennych, pasuje idealnie do kształtu pola, gdzie wpisane są zera.



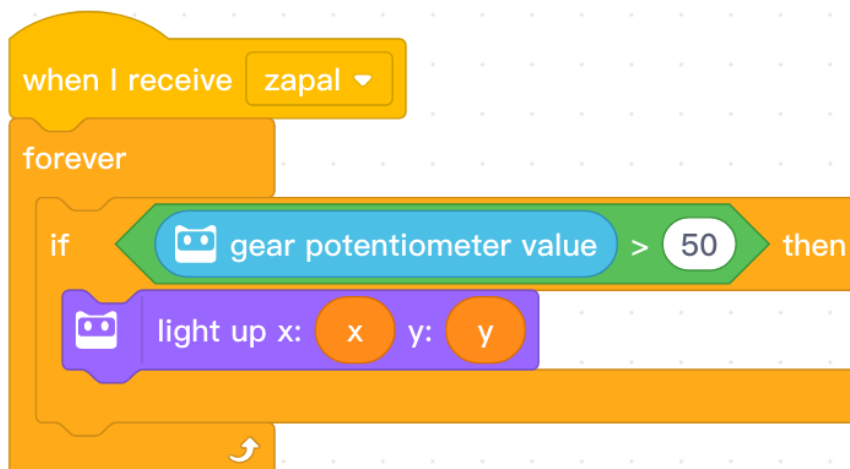
To rozwiązanie jest dobre i proste, ale ma jedną podstawową wadę. Za każdym razem zostanie zapalona dioda, na którą się przesuniesz. Jak temu zaradzić?

Codey rocky posiada pokrętko umiejscowione na boku. Możesz go użyć do decydowania czy zapalać diodę czy nie. Znow potrzebny będzie warunek *if* i pętla *forever* - zawsze.

*Zawsze*

*Jeżeli wartość potencjometru większa (>) od 50 wtedy*

*Zapal x: zmienna X y: zmienna Y*



W ten sposób, jeżeli potencjometr będzie ustawiony na minimum diody nie będą zapalane, a kiedy ustawisz go na maksimum spowoduje to zapalenie każdej diody, na którą się przesuniesz.

Teraz wystarczy połączyć robota *connect* i kliknąć *upload to device* i gotowe. Możesz rysować własne szlaczki i obrazki na ekranie robota za pomocą jego przycisków.

**Zadanie dodatkowe:**

- Zaprogramuj wyłączenie świecenia diod, czyli tak zwaną gumkę
- Zaprogramuj czyszczenie ekranu np. dwoma przyciskami wciśniętymi w tym samym czasie