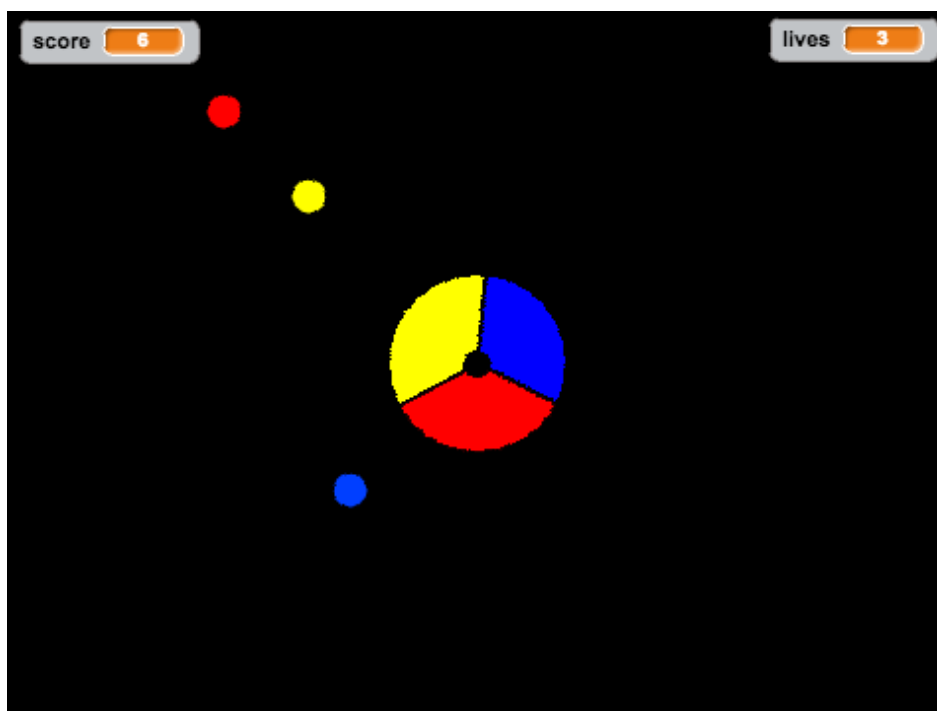


Każdy Klub Kodowania musi być zarejestrowany. Zarejestrowane kluby można zobaczyć na mapie na stronie codeclubworld.org - jeżeli nie ma tam twojego klubu sprawdź na stronie jumpton.cc/18CpLPy (ang.) co trzeba zrobić, by to zmienić.

Wstęp

Realizując ten projekt nauczysz się, jak stworzyć grę, w której musisz dopasować kolorowe kulki do odpowiedniej części pokrętki.



Zadania do wykonania

Wykonaj te **POLECENIA** krok po kroku



Przetestuj swój projekt

Kliknij na zieloną flagę, aby
PRZETESTOWAĆ swój kod



Zapisz swój projekt


Teraz **ZAPISZ** swój projekt

Krok 1: Budowa pokrętła

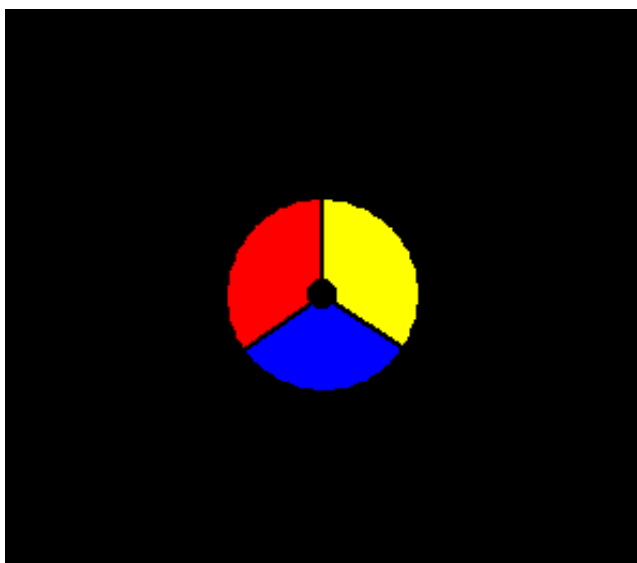
Zacznijmy od zbudowania pokrętła, które będzie używane do zbierania kulek.

✓ Zadania do wykonania

- Stwórz nowy projekt i usuń duszka-kota, aby projekt był pusty. ☐
- Aby wykonać ten projekt, powinieneś mieć katalog “Zasoby”, w którym znajdziesz obrazek pokrętła, który możesz użyć. Upewnij się, że masz ten katalog, a jeśli nie zapytaj o niego prowadzącego. ☐

 controller.png

- Stwórz nowego duszka korzystając z pliku “controller.png”, który znajdziesz w katalogu “Zasoby”. Jeśli nie masz tego obrazka możesz narysować własny! Zmień nazwę duszka na “pokrętło”. Pokoloruj też tło sceny na czarno. Całość powinna wyglądać mniej więcej tak: ☐



- Możesz poruszać pokrętłem w bardzo prosty sposób – obracając je w lewo lub w prawo kiedy gracz naciska ☐

strzałki:



- Przetestuj swoje pokrętło – powinno obracać się w lewo i w prawo. ☐
- Pomimo tego, że ten kod działa, byłoby znacznie lepiej, gdyby pokrętło stopniowo przyspieszało i zwalniało. Aby to zrobić, usuń kod pokrętła, który właśnie stworzyłeś i utwórz nową zmienną o nazwie `prędkość pokrętła`. ☐
- Dodaj poniższy kod do swojego pokrętła, aby używało zmiennej “prędkość pokrętła” do poruszania nim: ☐

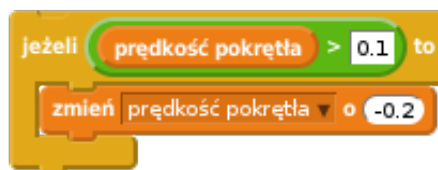


- Na tą chwilę powyższy kod nie będzie ruszał pokrętłem, ponieważ prędkość została ustawiona na 0! Zbuduj osobny skrypt w swoim pokrętle, aby zwiększać prędkość, kiedy naciśnięto strzałkę w prawo. ☐

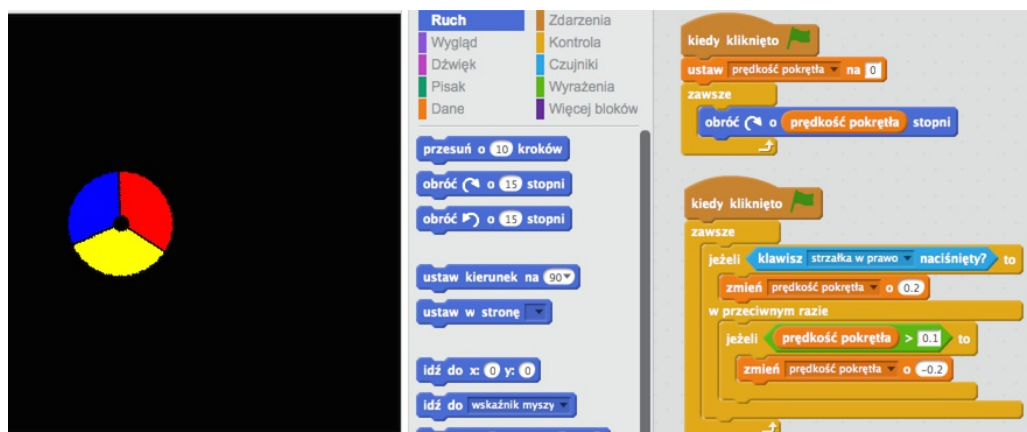


- Czy zauważyłeś puste miejsce w powyższym kodzie? Będziesz musiał dodać pewien kod by spowolnić pokrętło jeśli strzałka w prawo nie jest naciśnięta. Pamiętaj jednak, że zwalniać chcesz tylko do momentu, gdy prędkość pokrętła osiągnie 0, w przeciwnym razie zacznie się kręcić w drugą stronę.

Powinieneś dodać taki kod:



Pokrętło powinno wyglądać tak:



- Przetestuj ponownie swój projekt. Gdy trzymasz wciśnięty klawisz ze strzałką w prawo, twoje pokrętło powinno przyspieszać. Kiedy przestaniesz wciskać ten klawisz, pokrętło powinno stopniowo zwalniać.



Zapisz swój projekt

Wyzwanie: Obrót w lewo

Zduplikuj cały skrypt pokrętła odpowiedzialny za obracanie go w prawo. Czy potrafisz zmodyfikować ten zduplikowany kod w taki sposób, by pokrętło obracało się w lewo, kiedy wciśniesz strzałkę w lewo?

Będziesz musiał zmienić niektóre liczby w swoim kodzie! (Podpowiedź: pokrętło będzie obracać się w lewo jeśli zmienna `prędkość pokrętła` będzie miała wartość ujemną).



Zapisz swój projekt

Krok 2: Łapanie kropek

Dodajmy teraz do gry kropki, które gracz będzie musiał łapać obracając odpowiednio pokrętłem.

✓ Zadania do wykonania

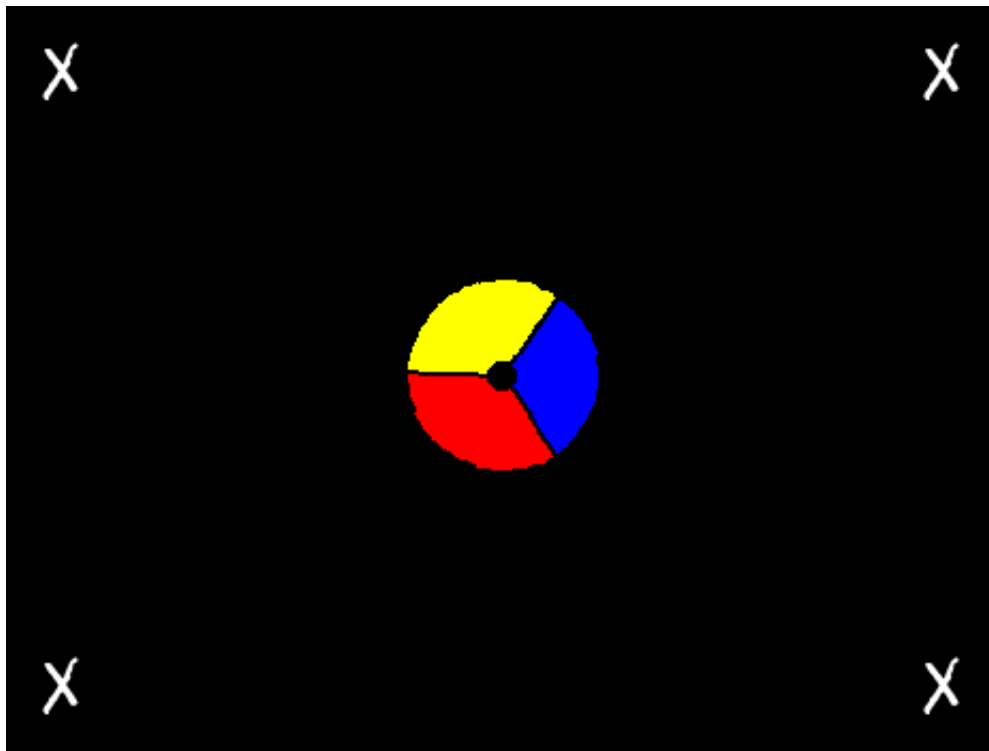
- Stwórz nowego duszka i nazwij go “czerwony”. Duszek ten powinien być małą czerwoną kropką. ☐



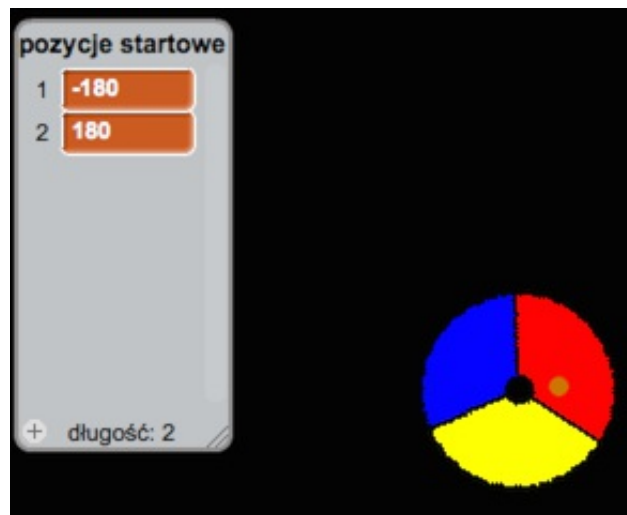
- Dodaj poniższy skrypt do czerwonej kropki, aby co kilka sekund powstawał nowy klon kropki: ☐



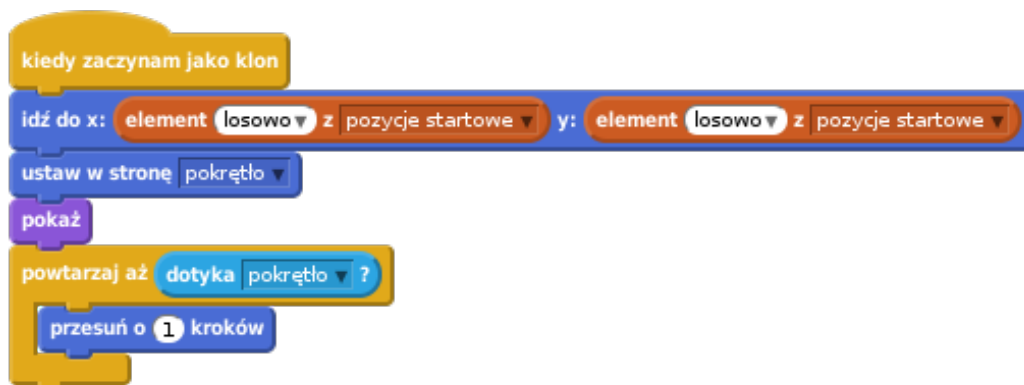
- Każdy klon po stworzeniu powinien pojawić się w jednym z czterech narożników sceny.



Aby to zrobić, najpierw utwórz nową listę i nazwij ją **pozycje startowe**, następnie kliknij na (+) i dodaj do listy wartości -180 i 180.



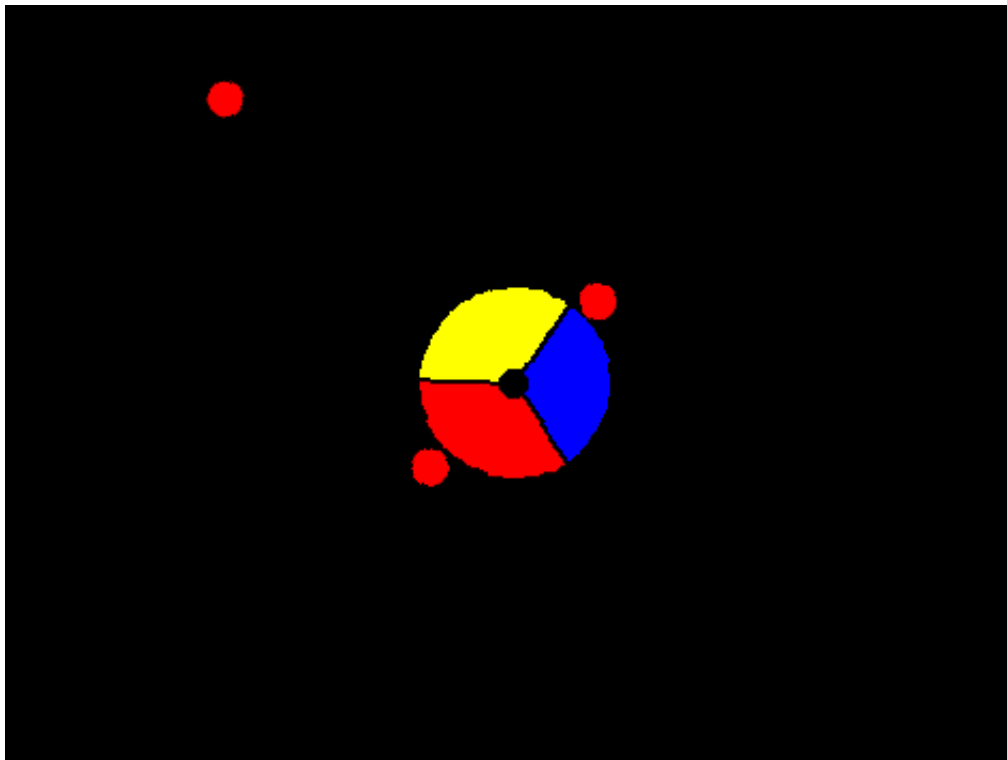
- Możesz użyć tych dwóch elementów listy do wybrania dowolnego narożnika sceny. Po dodaniu poniższego kodu do duszka-czerwonej kropki każdy klon pojawi się w losowym narożniku i będzie powoli leciał w kierunku pokręta.



Powyższy kod wybiera “-180” albo “180” dla pozycji x oraz y. To znaczy, że każdy klon będzie zaczynał w dowolnym narożniku sceny.

- Przetestuj projekt. Powinieneś zobaczyć mnóstwo czerwonych kropek, które pojawiają się w rogach ekranu i powoli lecą w stronę pokręta.

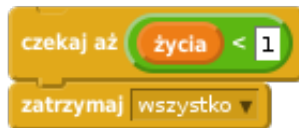




- Stwórz dwie nowe zmienne i nazwij je **życia** i **wynik**. ☐
- Dodaj kod do sceny, który na początku gry ustawi **życia** na 3, a **wynik** na 0. ☐
- Teraz musisz dodać kod do duszka-czerwonej kropki na końcu bloku **kiedy zaczynam jako klon**, który będzie albo dodawał 1 do **wyniku** jeśli kolory się zgadzają, albo odejmował 1 z **życ** gracza, jeśli kolory się nie zgadzają. ☐



- Po dodaniu tego kodu na końcu skryptu sceny gra zakończy się, kiedy gracz straci wszystkie życia: ☐



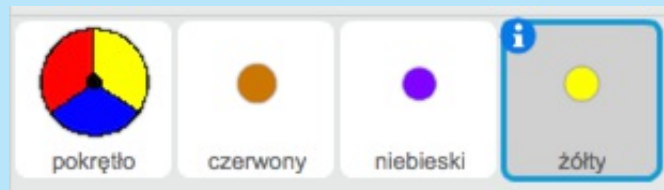
- Przetestuj swoją grę i upewnij się, że kod działa tak, jak się tego spodziewasz.



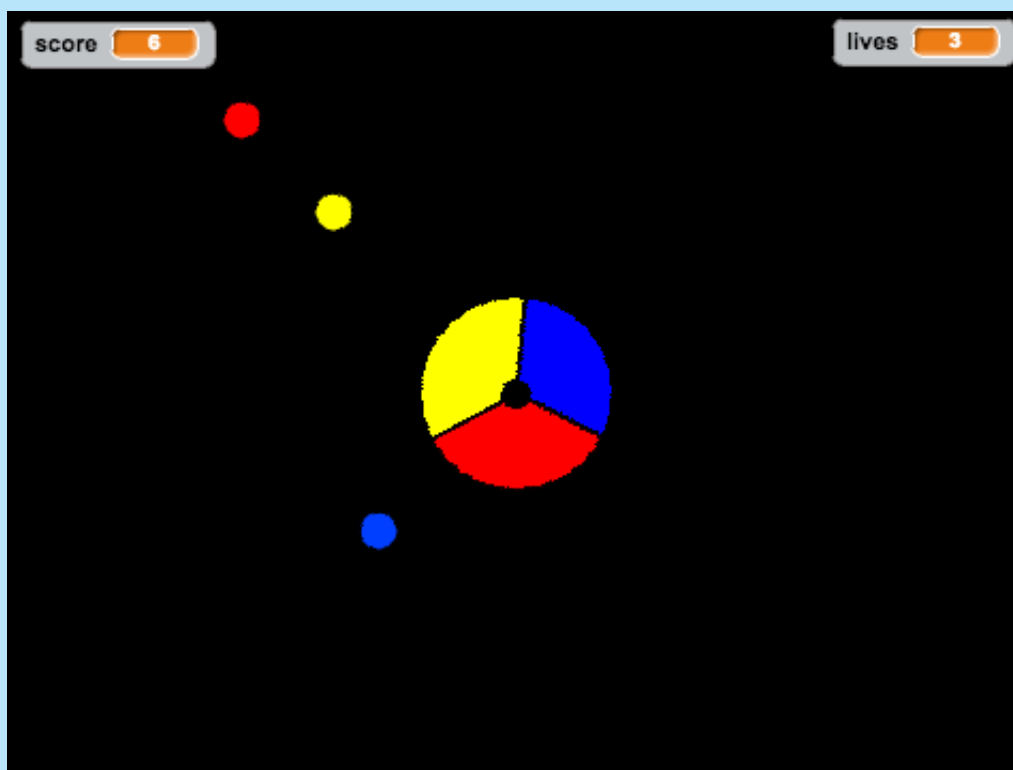
Zapisz swój projekt

Wyzwanie: Więcej kropek

Zduplikuj duszka-czerwoną kropkę dwa razy i nazwij nowe duszki “żółty” i “niebieski”.



Zmień te duszki (pamiętaj też zmienić ich kod) w taki sposób, by każdy kolor kropki pasował do koloru na pokrętle. Pamiętaj, aby przetestować swój projekt i upewnić się, że zdobywasz punkty i tracisz życia w odpowiednim momencie. Sprawdź też, czy gra nie jest przypadkiem za prosta albo za trudna.



Zapisz swój projekt

Krok 3: Zwiększanie trudności

Teraz sprawimy, by gra stawała się coraz trudniejsza w miarę upływu czasu. Zrobimy to zmniejszając opóźnienie z jakim pojawiają się kolejne kropki.

✓ Zadania do wykonania

- Stwórz nową zmienną i nazwij ją `opóźnienie`. ☐
- Dodaj nowy skrypt na scenie, który będzie ustawiał opóźnienie na wysoką liczbę, a później powoli ją zmniejszał. ☐



Zauważ, że w podobny sposób działa stoper!

- Teraz możesz użyć zmiennej `opóźnienie` w skryptach od czerwonej, żółtej i niebieskiej kropki. Usuń kod, który czeka losowy czas pomiędzy tworzeniem kolejnych klonów i zastąp go twoją zmienną `opóźnienie`: ☐



- Przetestuj nową zmienną `opóźnienie` i zobacz czy czas pomiędzy nowymi kropkami powoli spada. Czy działa to dla wszystkich trzech kolorów kropek? Widzisz, jak wartość `opóźnienia` spada? ☐



Zapisz swój projekt

Wyzwanie: Szybsze kropki

Możesz ulepszyć swoją grę dodając zmienną **prędkość**, która sprawi, że na początku kropki będą ruszać się za każdym razem o jeden krok, ale później będą poruszać się coraz szybciej i szybciej. Będzie to działać bardzo podobnie do tego, jak działa **opóźnienie**, którego przed chwilą użyliśmy.



Zapisz swój projekt

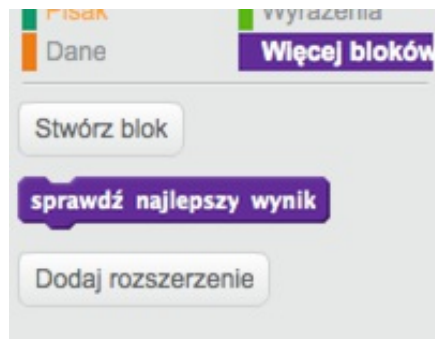
Krok 4: Najlepszy wynik

Zapiszmy najlepszy wynik, by gracze mogli rywalizować.



Zadania do wykonania

- Stwórz zmienną **najlepszy wynik**. ☐
- Kliknij na scenę i utwórz nowy blok, który nazwij **sprawdź najlepszy wynik**. ☐



- Tuż przed zakończeniem gry dodaj twój nowy blok. ☐



- Dodaj poniższy kod do twojego bloku, aby zapisać aktualny **wynik** jako **najlepszy wynik** **jeżeli** jest wyższy niż dotychczasowy:



- Przetestuj kod, który właśnie dodałeś. Zagraj i sprawdź, czy **najlepszy wynik** zapisuje się poprawnie.



Zapisz swój projekt

Wyzwanie: Ulepsz swoją grę!

Can you think of ways to improve your game? For example, you could create special dots that:

Masz pomysł jak można jeszcze bardziej ulepszyć grę?
Możesz stworzyć na przykład specjalne kropki, które:

- ☐ dają dwa razy więcej punktów,
- ☐ zwalniają kropki
- ☐ ukrywają wszystkie inne kropki na ekranie!



Wyzwanie: Game menu

Możesz też dodać menu (z przyciskami) do swojej gry. Możesz dodać stronę z instrukcją i osobny ekran, gdzie wyświetlisz najlepszy wynik. Jeśli nie pamiętasz jak to zrobić zajrzyć do materiałów z projektu “Brain Game”.