

2.1 Duszek w labiryncie

DOWIESZ SIĘ, JAK

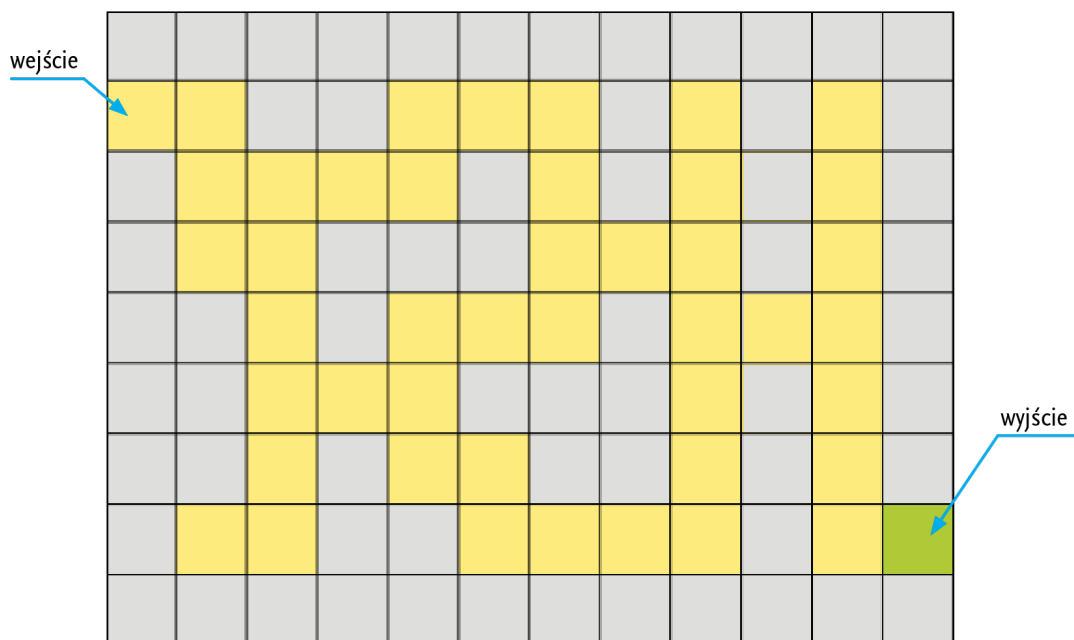
- wykorzystywać zdobyte do tej pory umiejętności do tworzenia gier,
- wybierać optymalne polecenia spośród dostępnych.

Kolejne cztery lekcje tego działu są poświęcone programowaniu w Scratchu gier komputerowych. Tym razem zrealizujesz projekt polegający na przeprowadzeniu duszka przez labirynt za pomocą klawiszy ze strzałkami.

SCENA

Duszek powinien poruszać się po labiryncie, który można przejść od startu do mety.

- Wykorzystaj plik dostarczony przez nauczyciela lub przygotuj w edytorze grafiki planszę o wymiarach 480 px × 360 px złożoną z kwadratowych pól o boku 40 px w kolorach szarym (ściany), żółtym (ścieżka) i zielonym (wyjście). Plik zapisz w formacie SVG lub PNG.



Rys. 1. Przykładowy labirynt

- Uruchom program Scratch. Wstaw plik tła – wybierz opcję **Wybierz tło** → **Wczytaj tło**.

STEROWANIE DUSZKIEM

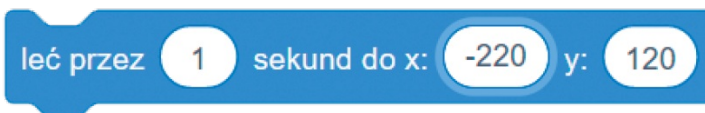
Teraz możesz się zająć oprogramowaniem duszka.

- Usuń kota i wybierz z biblioteki innego duszka – takiego, którego postać jest widoczna w widoku od góry.



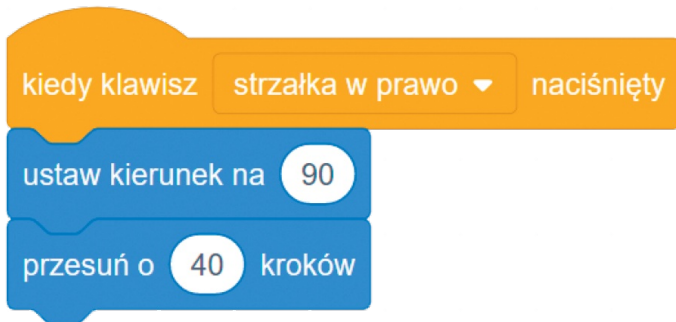
Rys. 2. Przykładowe duszki

- Dopasuj wielkość wybranego duszka do rozmiaru kratki – w polu **Rozmiar** wpisz odpowiednią wartość.
- Ułóż skrypt zielonej flagi, w którym ustalisz warunki początkowe duszka – wstaw bloki odpowiedzialne za ustawienie początkowe duszka.
- Rozważ dostępne bloki z grupy **Ruch**. Blok **przesuń o (...) kroków** przesuwa duszka o określoną odległość zgodnie z aktualnym kierunkiem, a duszek powinien trafić na konkretną „startową” pozycję. Bloki **idź do (...)** oraz **leć przez (...) sekund do (...)** przesuwiają duszka do losowej pozycji lub pozycji kursora myszy – w grę wchodziłaby więc druga opcja, ale duszek ma wracać na odpowiednie pole automatycznie. Z kolei polecenia **zmień i ustaw** dotyczą tylko jednej współrzędnej i trzeba by je łączyć. Dobrym wyborem byłoby polecenie **idź do x: (...) y: (...)**, które przesuwa duszka do punktu o określonych współrzędnych – tyle że duszek tam po prostu skacze. Blok **leć przez (...) sekund do x: (...) y: (...)** gwarantuje płynne przemieszczanie się po linii prostej przez określony czas z punktu, w którym aktualnie duszek się znajduje, do miejsca oznaczonego jako punkt docelowy.



Rys. 3. Blok odpowiedzialny za pozycję duszka

- Teraz zaprogramuj sterowanie duszkiem za pomocą klawiszy ze strzałkami – np. po naciśnięciu klawisza ze strzałką skierowaną w prawo duszek powinien pójść w stronę prawego brzegu ekranu, a po naciśnięciu klawisza ze strzałką skierowaną do góry – w górę.
- Ułóż polecenia do wykonania po naciśnięciu klawisza ze strzałką w prawo – blok **ustaw kierunek na 90** oznacza, że duszek będzie się poruszał w poziomie w prawo. Aby przeskoczyć do kolejnego pola, musi zrobić 40 kroków.

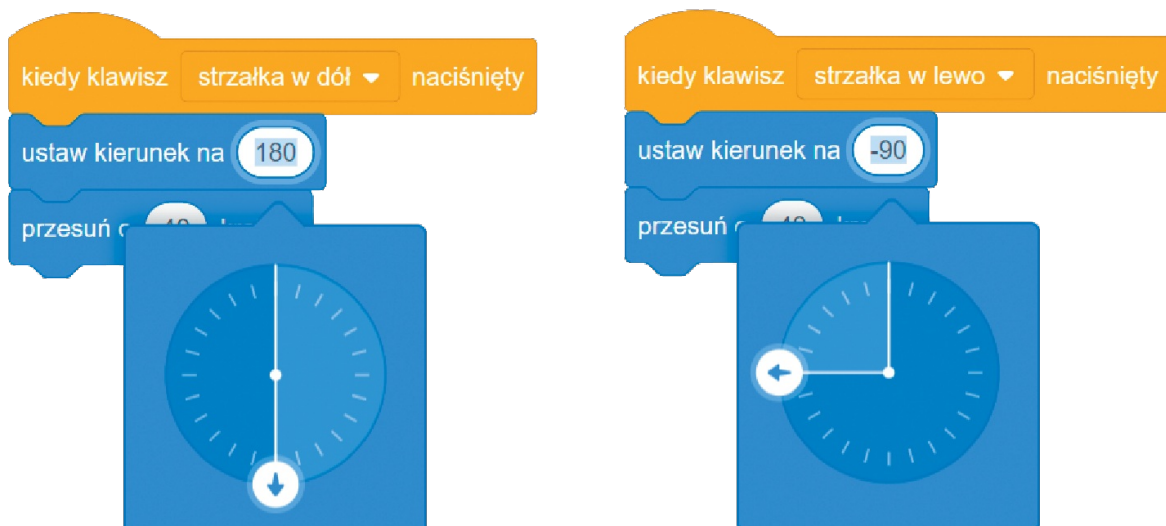


Rys. 4. Skrypt przesuwania duszka w prawo

- Sprawdź skrypt. Jeśli działa poprawnie, zduplikuj go i odpowiednio zmodyfikuj, aby duszek mógł przesuwać się w pozostałych kierunkach.

Jak ustalić kierunek

Podczas uzupełniania bloku **ustaw kierunek na (...)** można wpisać w pole wartości stopnie, jednak trzeba pamiętać o tym, że program odpowiednio je przekształci, np. 270° zamieni się na -90 . Można też skorzystać z „zegara” – przesuwać wskazówkę i sprawdzać, jak zmienia się kierunek.



- Zaprogramuj poruszanie się duszka po trasie labiryntu – duszek powinien chodzić tylko po żółtych polach, szare kwadraty są ścianami labiryntu.

- Utwórz nowy blok **kolory**, który będzie badać kolory kwadratów, i odpowiednio go zdefiniuj.
- Wstaw blok warunkowy **jeżeli (...) to (...)**, którego zadaniem będzie sprawdzenie, czy duszek dotyka ściany (szarego koloru) oraz cofnięcie go na ścieżkę (poprzednie pole), jeśli takie zdarzenie wystąpi.



Rys. 5. Pierwsza część bloku sprawdzania koloru kwadratu

- Dodaj kolejny blok warunkowy, który opisuje następującą sytuację – jeżeli duszek dotyka koloru zielonego, to podaje informację o tym, że doszedł do końca labiryntu. Wówczas odgrywany jest dźwięk (wybierz oklaski lub inny dźwięk z biblioteki dźwięków), a następnie duszek skacze na pozycję początkową.



Rys. 6. Druga część bloku sprawdzania koloru kwadratu

- Do każdego skryptu przesuwania duszka dodaj wywołanie bloku sprawdzania kolorów.



Rys. 7. Skrypt przesuwania duszka w prawo ze sprawdzaniem koloru następnego kwadratu

- Na koniec uzupełnij skrypty blokiem **zatrzymaj (...)**.
- Blok **zatrzymaj ten skrypt** powoduje zatrzymanie wykonywania danego skryptu. Częstym błędem jest wybranie opcji **wszystko** – polecenie to działa tak jak kliknięcie czerwonego przycisku **Zatrzymaj**, tj. powoduje zatrzymanie wszystkich skryptów wszystkich duszków i sceny. Z kolei blok **zatrzymaj** z opcją **inne skrypty duszka** zatrzymuje działanie skryptów danego duszka – poza skryptem, w którym został uruchomiony blok **zatrzymaj (...)**.

ZADANIA

1. Do projektu zrealizowanego podczas lekcji wprowadź zliczanie błędnych ruchów duszka.
2. Do skryptu, w którym duszek próbuje wejść na szare pole, dodaj dźwięk ostrzegawczy.

