Temat 7. Tworzymy gry komputerowe w języku Scratch

1. Realizujemy sytuacje warunkowe – sterujemy duszkiem
2. Tworzymy grę dla jednego gracza
3. Stosujemy zmienne – zliczamy punkty
4. Dodajemy warunki zakończenia gry
5. Tworzymy grę dla dwóch graczy
6. Dodajemy kolejne poziomy gry

Warto powtórzyć

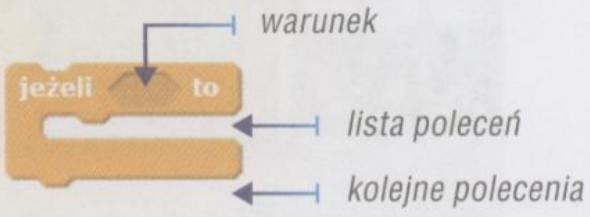
1. Czym jest język programowania?
2. Na czym polega tworzenie programu w języku Scratch?
3. W jaki sposób zapisywaliśmy wielokrotnie powtarzające się polecenia w języku Scratch?
4. W jaki sposób można uruchomić program zapisany w języku Scratch?
5. Jakie wartości przyjmują współrzędne *x* i *y* sceny w programie Scratch?
6. W jaki sposób zmieniamy postać duszka i tło sceny?

1. Realizujemy sytuacje warunkowe – sterujemy duszkiem

Chcemy utworzyć program, w którym duszek będzie rysował różne figury, zależnie od naciśnięcia wybranego klawisza, np. strzałki w lewo.

Jeśli wynik działania programu zależy od spełnienia warunku (np. naciśnięcia klawisza strzałki w lewo), mamy do czynienia z **sytuacją warunkową**.

Aby w języku Scratch zrealizować sytuację warunkową, możemy użyć polecenia **jeżeli** z grupy **Kontrola**.



Jeśli *warunek* **jest spełniony**, realizowana jest *lista poleceń* programu umieszczona po słowie **to**.

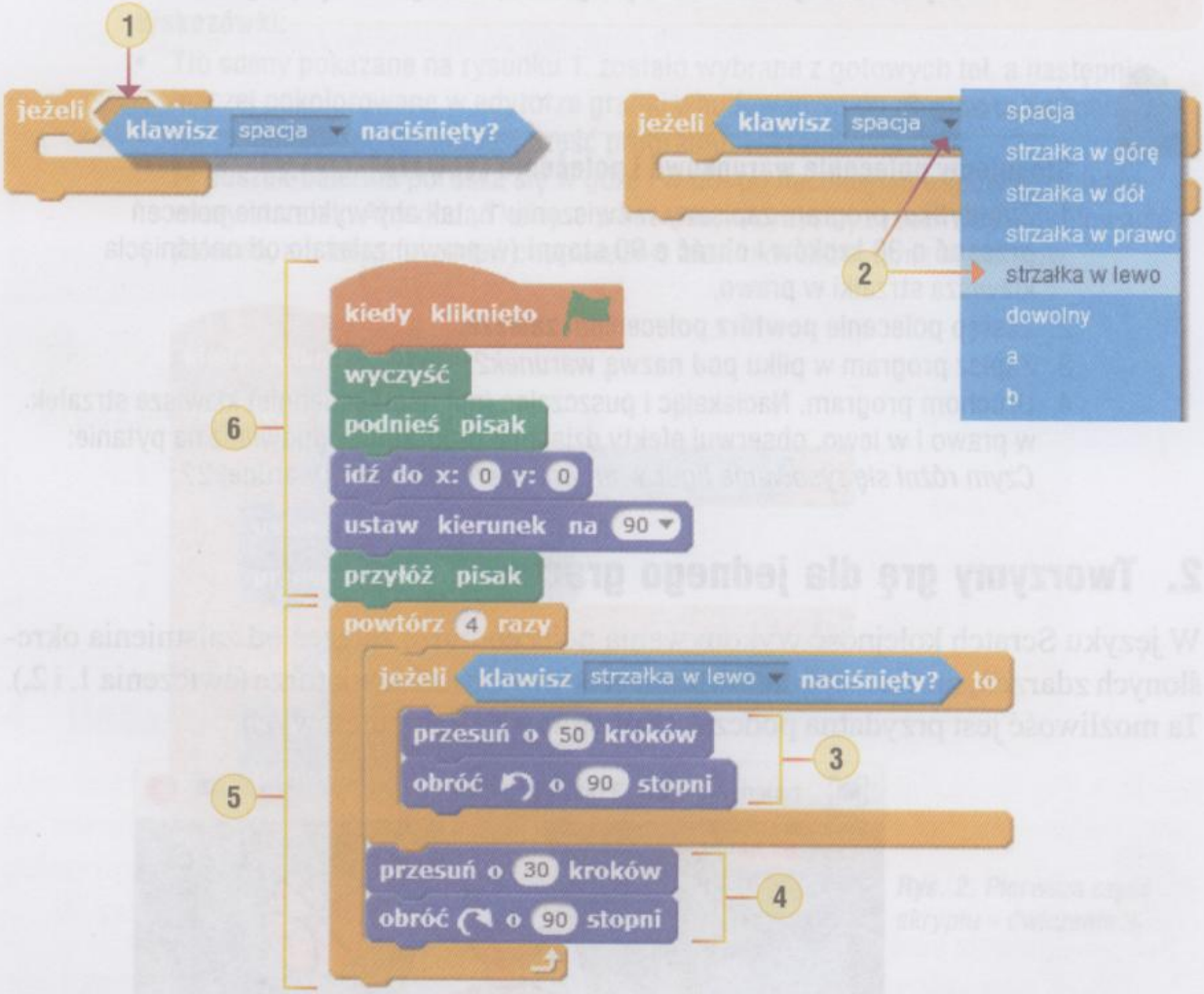
Następnie (niezależnie od spełnienia *warunku*) realizowane są *kolejne polecenia* programu.

Uwaga: Jest to uproszczona wersja polecenia warunkowego.

Przykład 1.

Stosowanie polecenia warunkowego w wersji uproszczonej

1. Umieść *warunek* w polu warunku – *warunkiem* może być wybrane zdarzenie z grupy **Czujniki**, np. **klawisz** *spacja* **naciśnięty**?
2. Rozwiń listę wyboru polecenia (naciskając strzałkę przy domyślnym klawiszu *spacja*) i wybierz *strzałka w lewo*.
3. Po słowie **to** umieść polecenia, które mają być wykonywane, jeśli *warunek* jest spełniony.
4. Poniżej elementu **jeżeli** umieść *kolejne polecenia* programu.
5. Zastosuj polecenie **powtórz**, aby czterokrotnie powtarzać wybrane polecenia.
6. Uzupełnij program, dodając polecenie **kiedy** i po nim pozostałe polecenia programu.



Uwagi:

- Jeśli klawisz *strzałka w lewo* nie jest naciśnięty, realizowane są polecenia **przesuń o** *30* **kroków** i **obróć o** *90* **stopni** (w prawo), które powtarzane są cztery razy.
- Jeśli klawisz *strzałka w lewo* jest naciśnięty, czyli *warunek* jest spełniony, realizowane są polecenia **przesuń o** *50* **kroków** i **obróć o** *90* **stopni** (w lewo), a następnie – polecenia **przesuń o** *30* **kroków** i **obróć o** *90* **stopni** (w prawo). Wszystkie polecenia powtarzane są cztery razy.

Ćwiczenie 1.

Stosujemy polecenie warunkowe „jeżeli" i polecenie „powtórz"

1. Ułóż polecenia pokazane w przykładzie 1.
2. Zapisz program w pliku pod nazwą *warunek1.*
3. Uruchom program. Odpowiedz na pytanie: *Jaką figurę narysował duszek?*
4. Naciśnij klawisz strzałki w lewo i, trzymając wciśnięty, uruchom ponownie program. Odpowiedz na pytanie: *Jaką teraz figurę narysował duszek?*

Aby wybrane polecenia były wykonywane nieskończenie wiele razy, można zastosować polecenie **zawsze** z grupy **Kontrola**.

Aby zakończyć działanie programu, należy kliknąć przycisk .

Ćwiczenie 2.

Stosujemy polecenie warunkowe i polecenie „zawsze"

1. Zmodyfikuj program zapisany w ćwiczeniu 1., tak aby wykonanie poleceń **przesuń o** 30 **kroków** i **obróć o** 90 **stopni** (w prawo) zależało od naciśnięcia klawisza strzałki w prawo.
2. Zastąp polecenie **powtórz** poleceniem **zawsze**.
3. Zapisz program w pliku pod nazwą *warunek2.*
4. Uruchom program. Naciskając i puszczając (np. naprzemiennie) klawisze strzałek w prawo i w lewo, obserwuj efekty działania programu. Odpowiedz na pytanie: *Czym różni się rysowanie figur w programach* warunek1 *i* warunek2*?*

2. Tworzymy grę dla jednego gracza

W języku Scratch kolejność wykonywania poleceń może zależeć od zaistnienia określonych zdarzeń, np. naciśnięcia wybranego klawisza na klawiaturze (ćwiczenia 1. i 2.). Ta możliwość jest przydatna podczas tworzenia gier komputerowych.

Rys. 1. *Przykładowa scena gry – ćwiczenie 3.*

Utworzymy grę, w której wezmą udział dwa duszki (rys. 1.). Jeden z duszków będzie poruszał się losowo po całej scenie. Drugim duszkiem będziemy sterować, naciskając klawisze strzałek. Jeśli duszki dotkną się, otrzymujemy 1 punkt. W kolejnych ćwiczeniach będziemy modyfikować grę, m.in. dodamy warunki zakończenia gry.

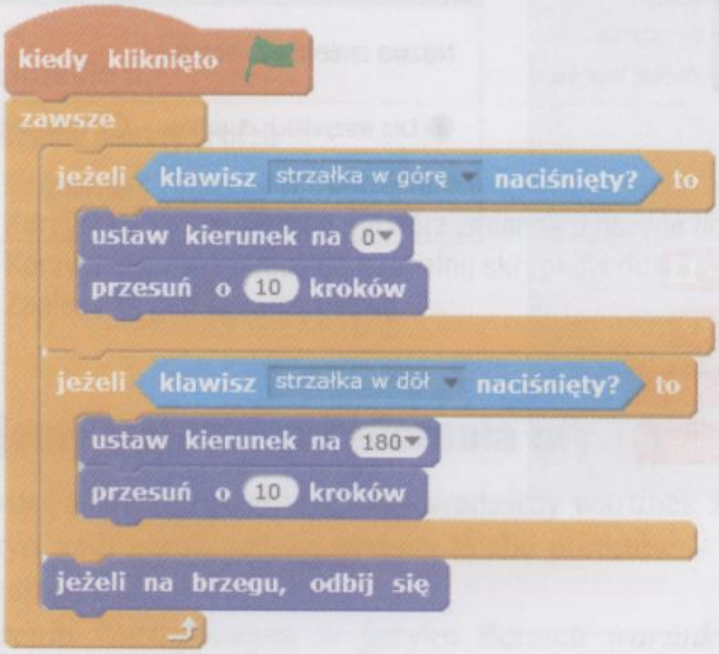
Ćwiczenie 3.

Sterujemy duszkiem w czterech kierunkach

1. Utwórz program, w którym scena zostanie zamieniona na wybrany obraz, a duszek-balerina będzie poruszał się po scenie w prawo, w lewo, w górę i w dół, zależnie od naciśniętych klawiszy strzałek.
2. Wszystkie polecenia powtarzaj nieskończenie wiele razy.
3. Zapisz program w pliku pod nazwą *balerina z piłką*.

Wskazówki:

- Tło sceny pokazane na rysunku 1. zostało wybrane z gotowych teł, a następnie inaczej pokolorowane w edytorze grafiki wbudowanym do programu Scratch.
- Na rysunku 2. pokazana jest część programu, w której określono, że duszek-balerina porusza się w górę i w dół po naciśnięciu odpowiednich klawiszy strzałek. Aby dodać kolejne polecenia, skorzystaj z możliwości dublowania poleceń, a następnie zmień odpowiednio *warunki* i ustawienia kierunków.



***Rys. 2.** Pierwsza część skryptu – ćwiczenie 3.*

Ćwiczenie 4.

Piszemy skrypt dla duszka poruszającego się losowo po całej scenie

1. Otwórz plik zapisany w ćwiczeniu 3.
2. Dodaj duszka-piłkę i utwórz dla niego skrypt, w którym duszek będzie poruszał się nieskończenie wiele razy w sposób losowy po całej scenie.
3. Zapisz plik pod tą samą nazwą.

Wskazówka: Zastosuj polecenie **leć** i umieść je w pętli **zawsze**.

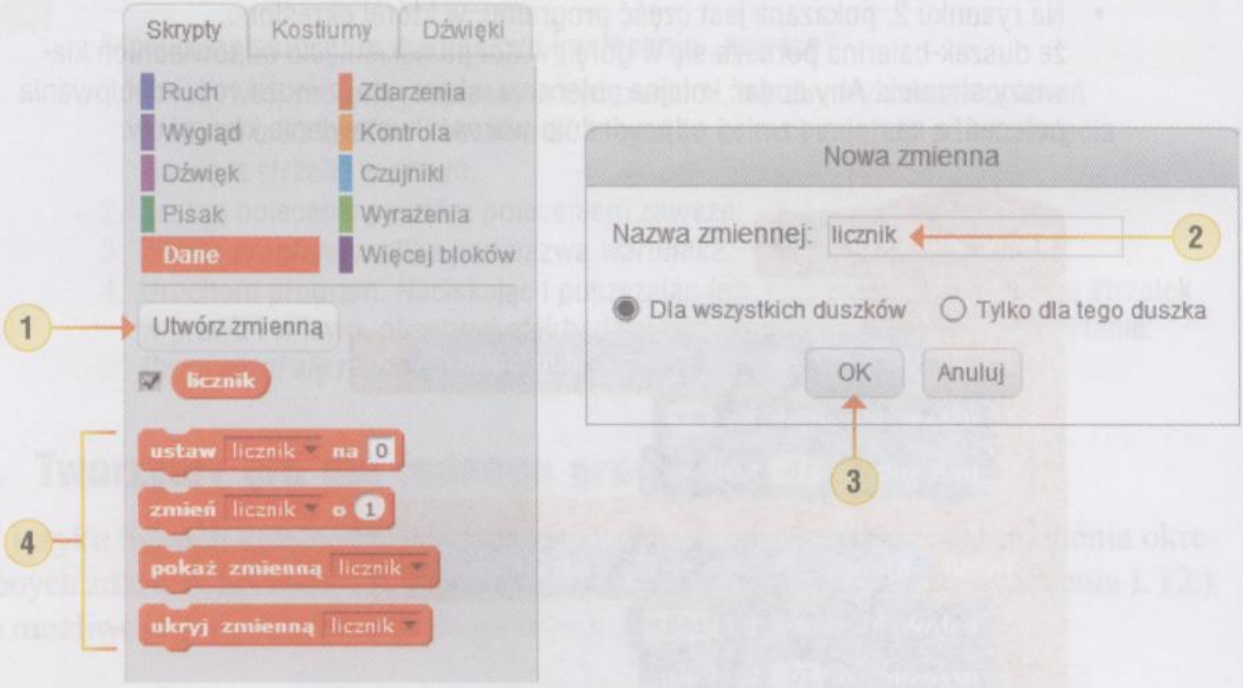
3. Stosujemy zmienne – zliczamy punkty

W programie Scratch można korzystać ze zmiennych. W naszej grze utworzymy jedną zmienną o nazwie *licznik*, dzięki której będziemy pamiętać liczbę zdobytych punktów.

Przykład 2.

Tworzenie nowej zmiennej w programie

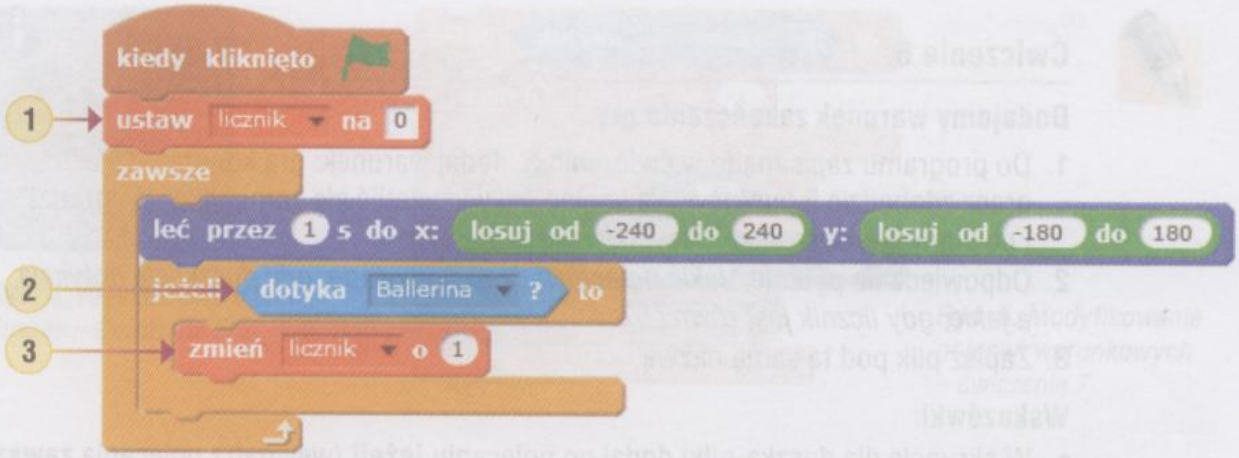
1. Wybierz element Utwórz zmienną z grupy **Dane** – otworzy się okno dialogowe **Nowa zmienna**.
2. W polu tekstowym **Nazwa zmiennej** wpisz nazwę zmiennej, np. *licznik*.
3. Kliknij przycisk **OK**.
4. Utworzona zmienna i polecenia dotyczące jej stosowania pojawią się w panelu poleceń.

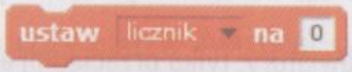


Przykład 3.

Zliczanie punktów zależne od warunku (zastosowanie polecenia „jeżeli”)

1. Aby punkty w grze zliczały się od zera, wyzeruj na początku programu *licznik*, stosując polecenie ustaw licznik na 0.
2. Aby zliczać punkty, gdy duszki się dotkną, do polecenia warunkowego wstaw warunek dotyka ? z grupy **Czujniki** i z listy wyboru wybierz nazwę danego duszka.
3. Jeśli warunek jest spełniony, powinny zliczać się punkty. Dodaj polecenie zmień licznik o 1.

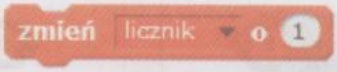


Uwaga: *Licznik* po wykonaniu polecenia 

jest równy zero.

Wykonanie pierwszy raz polecenia

spowoduje, że *licznik* przyjmie wartość 1.

Kolejne wykonanie polecenia 

powoduje, że *licznik* zwiększa się o jeden, czyli będzie przyjmował wartości: 2, 3, 4 itd.

Ćwiczenie 5.

Zliczamy punkty w grze

1. Otwórz plik zapisany w ćwiczeniu 4.
2. Korzystając z przykładu 2., utwórz zmienną o nazwie *licznik*.
3. Korzystając z przykładu 3., uzupełnij skrypt dla duszka-piłki.
4. Zapisz plik pod tą samą nazwą.

4. Dodajemy warunki zakończenia gry

Aby zatrzymać zliczanie punktów, wprowadzimy warunek zakończenia gry. Gra ma się zakończyć, jeśli gracz uzyska podaną liczbę punktów – w tym celu zastosujemy polecenie warunkowe.

W poleceniu warunkowym w języku Scratch *warunkiem* może być wyrażenie logiczne reprezentowane przez elementy z grupy **Wyrażenia**, np.

. W polach tekstowych elementu można umieszczać m.in.: nazwę zmiennej, liczbę, literę, np.

.

Ćwiczenie 6.

Dodajemy warunek zakończenia gry

1. Do programu zapisanego w ćwiczeniu 5. dodaj warunek: gra kończy się, jeśli gracz zdobędzie 5 punktów. Na koniec ma wyświetlić się komunikat „Wygrana!" i powinny się zatrzymać wszystkie skrypty.
2. Odpowiedz na pytanie: *Jakie polecenia są wykonywane, gdy duszki się dotykają, a jakie, gdy licznik jest równy 5?*
3. Zapisz plik pod tą samą nazwą.

Wskazówki:

- W skrypcie dla duszka-piłki dodaj po poleceniu **jeżeli** (wewnątrz polecenia **zawsze**) polecenie warunkowe pokazane na rysunku 3.
- Tworząc warunek w poleceniu **jeżeli**, nie wpisuj nazwy zmiennej *licznik* z klawiatury, tylko przeciągnij element

do odpowiedniego pola tekstowego elementu

.

- Polecenie **zatrzymaj wszystko** zatrzymuje wszystkie skrypty dla wszystkich duszków.

***Rys. 3.** Warunek zakończenia gry – ćwiczenie 6.*

Ćwiczenie 7.

Dodajemy trzeciego duszka i wprowadzamy odejmowanie punktów

1. Otwórz plik zapisany w ćwiczeniu 6.
2. Dodaj duszka-nietoperza (rys. 4.), który, podobnie jak piłka, ma się poruszać losowo po całym ekranie.
3. Wprowadź warunek odejmowania punktów w grze – gdy duszek-balerina i duszek-nietoperz dotkną się, *licznik* ma zmniejszać się o jeden.
4. Wprowadź warunek zakończenia gry: gra kończy się, gdy *licznik* jest równy -5. Na koniec ma się wyświetlić komunikat „Przegrana!" i gra ma się zakończyć.
5. Zapisz program w pliku pod nazwą *gra-trzy duszki*.

Wskazówka: Polecenia dla duszka-nietoperza są bardzo podobne do poleceń dla duszka-piłki, dlatego można zdublować polecenia duszka-piłki i kopię przeciągnąć na duszka-nietoperza w obszarze **Duszki**. Następnie należy wykonać niezbędne poprawki (rys. 5.) i usunąć zerowanie zmiennej *licznik* w skrypcie duszka-nietoperza.

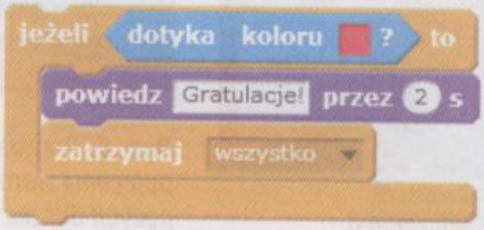
Rys. 4. Przykładowy trzeci duszek – ćwiczenie 7.

Rys. 5. Modyfikowanie poleceń warunkowych – ćwiczenie 7.

5. Tworzymy grę dla dwóch graczy

Chcemy utworzyć grę dla dwóch graczy – każdy będzie sterował swoim duszkiem, używając wybranych klawiszy. Warunkiem zakończenia gry, będzie dotknięcie koloru podanego w elemencie

z grupy **Czujniki** umieszczonym w poleceniu warunkowym (rys. 6.).



Rys. 6. Przykład zastosowania polecenia warunkowego do określenia warunku zakończenia gry – ćwiczenie 8.

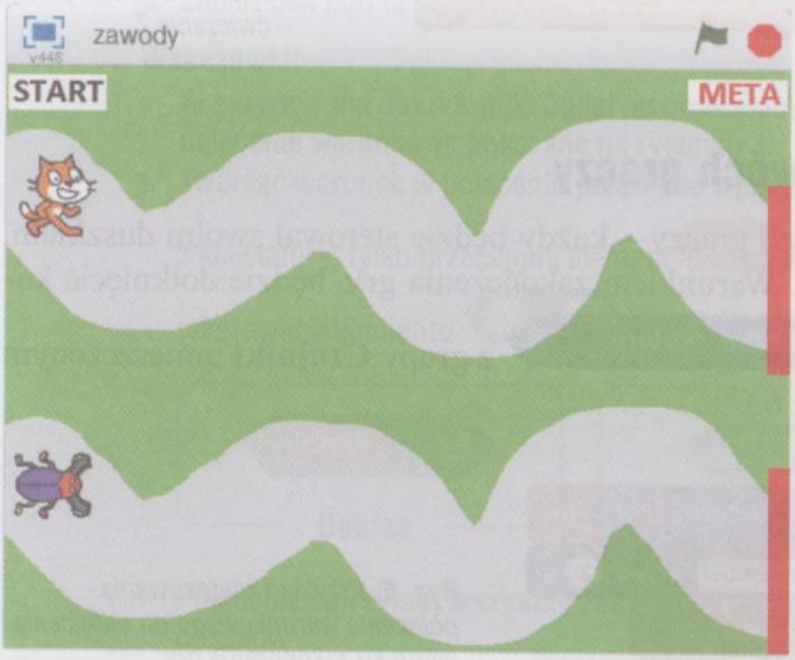
Ćwiczenie 8.

Tworzymy grę dla dwóch graczy

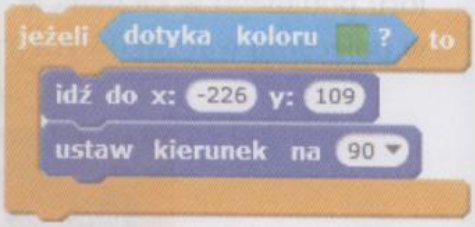
1. Utwórz grę, w której dwa duszki będą biegły po takich samych torach od startu do mety.
2. Wybierz postacie duszków z **Biblioteki duszków**. Korzystając z wbudowanego edytora grafiki, narysuj tło. Pamiętaj, żeby oba tory były takie same (wzór na rysunku 7.).
3. Dla każdego duszka utwórz oddzielny skrypt.
   a) Na początku każdego skryptu ustaw odpowiednio pozycję startową duszka, korzystając ze współrzędnych; ustaw kierunek duszka w prawo.
   b) Duszki mają mieć możliwość ruchu po torze w czterech kierunkach (w prawo, w lewo, w górę, w dół), zależnie od naciśniętych klawiszy. Dla każdego duszka wybierz cztery różne klawisze według własnego pomysłu.
   c) Jeśli duszek wyjdzie poza tor (dotknie tła koloru zielonego), powinien wrócić na start.
   d) Wygrywa ten duszek, który pierwszy dobiegnie do mety, czyli dotknie koloru czerwonego (rys. 7.). Na koniec ma wyświetlić się komunikat „Gratulacje!" i powinny się zatrzymać wszystkie skrypty.
   e) Polecenia b), c) i d) powtarzaj nieskończenie wiele razy.
4. Zapisz program w pliku pod nazwą *zawody* i zaproś koleżankę lub kolegę do wspólnej zabawy ☺.

Wskazówki:

- Zauważ, że skrypty dla każdego z duszków są prawie takie same. Wystarczy ułożyć skrypt dla jednego duszka, zdublować i upuścić na drugiego duszka. Następnie należy tylko zmienić wartości współrzędnych pozycji początkowej.
- We wszystkich skryptach należy umieścić polecenie warunkowe umożliwiające powrót na start (rys. 8.), określając odpowiednio współrzędne każdego z duszków na starcie.
- Aby gracze mieli wyrównane szanse w grze, obydwa duszki powinny być mniej więcej tej samej wielkości.



Rys. 7. Przykładowa scena do gry zawody *– ćwiczenie 8.*



Rys. 8. Przykład zastosowania polecenia warunkowego do zmiany pozycji duszka – ćwiczenie 8.

6. Dodajemy kolejne poziomy gry

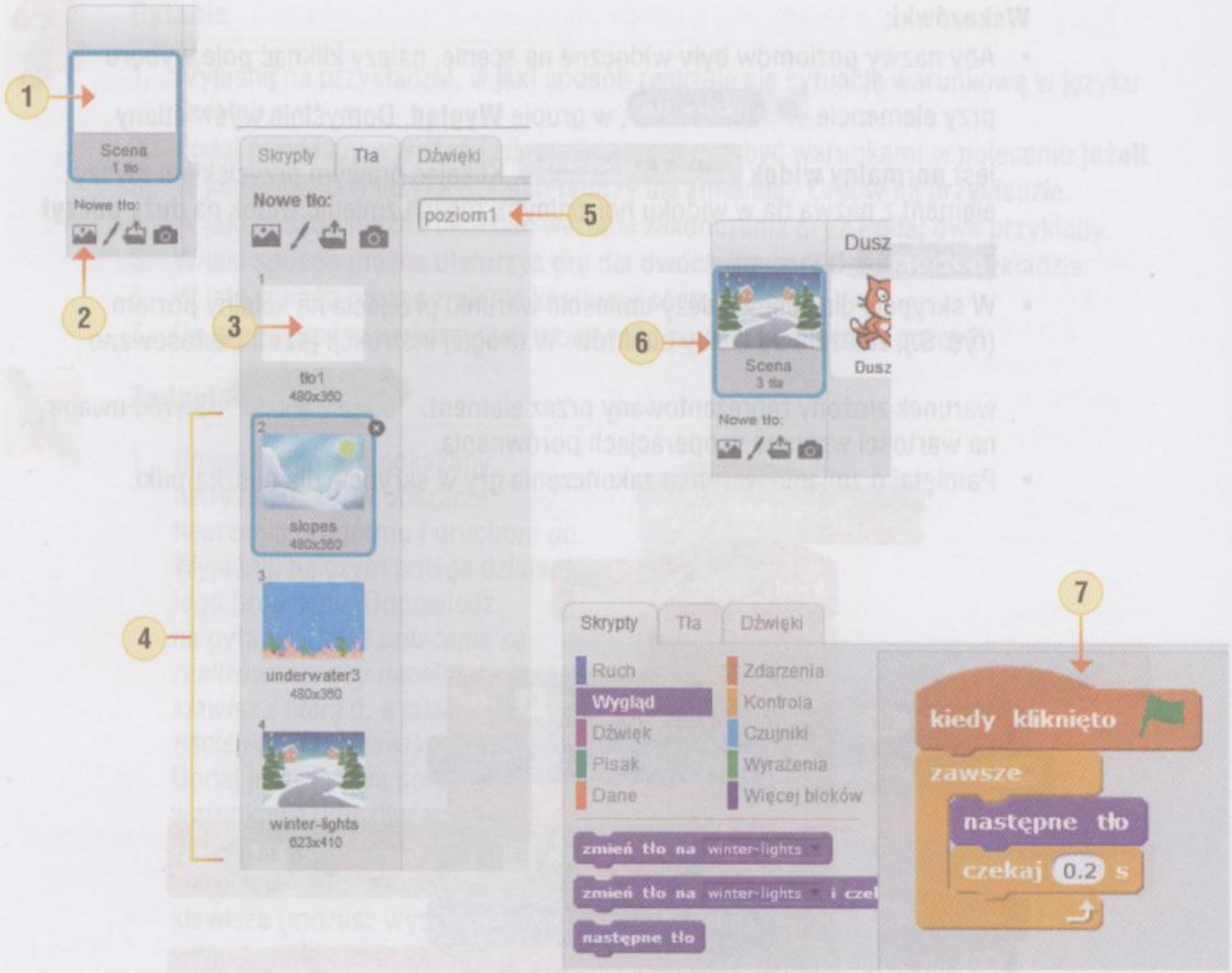
W programie Scratch możemy zmienić tło sceny – w podobny sposób, jak kostiumy duszków. Aby kontrolować zmianę tła, możemy napisać skrypt dla sceny.

Utworzymy grę, w której do kolejnych poziomów będziemy przechodzić, wykorzystując zmianę kostiumów sceny. Na wyższym poziomie, poza zmianą tła, można wprowadzić różne utrudnienia i ograniczenia, np. określać odpowiednio punktację, zwiększać szybkość poruszania się duszków, kontrolować czas gry itp.

Przykład 4.

Zmienianie kostiumów sceny

1. Kliknij ikonę **Scena**.
2. Kliknij przycisk **Nowe tło**.
3. Usuń scenę domyślną.
4. Dodaj trzy nowe tła z **Biblioteki teł**.
5. Zmień ich nazwy na: *poziom1*, *poziom2*, *poziom3*.
6. Przejdź do karty **Skrypty** i sprawdź, czy scena jest aktywna – na ikonie sceny powinno być widoczne niebieskie obramowanie.
7. W obszarze tworzenia programu napisz skrypt dla sceny.



Uwaga: Nie wszystkie polecenia są dostępne dla sceny, np. nie ma możliwości zastosowania poleceń z grupy **Ruch**.

Ćwiczenie 9.

Piszemy skrypt dla sceny – zmieniamy kostiumy

1. Korzystając z przykładu 4., napisz program, w którym tło sceny będzie się zmieniać co 0,2 sekundy.
2. Zapisz program w pliku pod nazwą *trzy tła*.

Wskazówka: W poleceniu **czekaj** czas 0,2 sekundy zapisujemy z kropką, czyli 0.2.

Ćwiczenie 10.

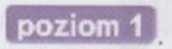
Modyfikujemy grę, dodając kolejne poziomy

1. Otwórz plik zapisany w ćwiczeniu 7.
2. Zmodyfikuj grę, dodając dwa nowe tła. Zmień nazwy teł na: *poziom1*, *poziom2*, *poziom3*. Dodaj wyświetlanie nazw poziomów ma scenie.
3. Na poziom drugi powinno się przechodzić po zdobyciu 5 punktów, na trzeci – po zdobyciu kolejnych 5 punktów. Gra ma się zakończyć po zdobyciu w sumie 15 punktów.
4. Zapisz program pod nazwą *gra-trzy poziomy*.

Wskazówki:

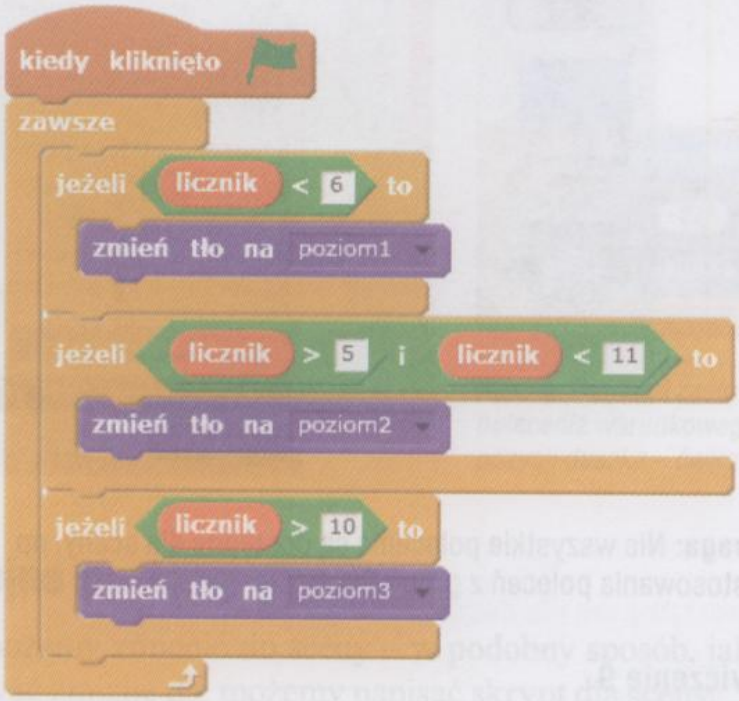
- Aby nazwy poziomów były widoczne na scenie, należy kliknąć pole wyboru przy elemencie

, w grupie **Wygląd**. Domyślnie wyświetlany jest **normalny widok**

. Klikając prawym przyciskiem myszy element z nazwą tła w widoku normalnym, można zmienić widok na **duży odczyt** 

.
- W skrypcie dla sceny należy umieścić warunki przejścia na kolejny poziom (rys. 9.), zależnie od liczby punktów. W drugiej instrukcji **jeżeli** zastosowano warunek złożony reprezentowany przez element:

. Zwróć uwagę na wartości wpisane w operacjach porównania.
- Pamiętaj o zmianie warunku zakończenia gry w skrypcie dla duszka-piłki.



***Rys. 9.** Skrypt dla sceny – ćwiczenie 10.*

Warto zapamiętać

- W języku Scratch sytuację warunkową realizujemy, stosując polecenie warunkowe, które w wersji uproszczonej ma postać: **jeżeli** *warunek* **to** *lista poleceń*.
- *Warunkiem* w poleceniu **jeżeli** może być wyrażenie logiczne (np. *punkty* >10, *licznik* = 20) lub zdarzenie z grupy **Czujniki** (np. **klawisz... naciśnięty?**, **dotyka ... ?, dotyka koloru ...?**).
- W języku Scratch można stosować zmienne. Zmienną należy wcześniej utworzyć.
- W języku Scratch możemy tworzyć gry dla jednego lub kilku graczy.
- Polecenia warunkowe możemy stosować do określania warunków zakończenia gry.
- Dla sceny w języku Scratch możemy również napisać skrypt.
- Tło sceny możemy zmienić w podobny sposób, jak kostiumy duszków.

Pytania

1. Wyjaśnij na przykładzie, w jaki sposób realizuje się sytuację warunkową w języku Scratch.
2. Podaj przykłady wyrażeń i zdarzeń, które mogą być warunkami w poleceniu **jeżeli**.
3. W jaki sposób w języku Scratch tworzy się zmienną? Omów na przykładzie.
4. W jaki sposób można określać warunki zakończenia gry? Podaj dwa przykłady.
5. W jaki sposób można utworzyć grę dla dwóch graczy? Omów na przykładzie.
6. W jaki sposób można zmienić kostiumy sceny?
7. Do czego można wykorzystać możliwość zmiany kostiumów sceny?

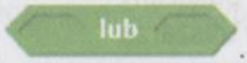
Zadania

1. Umieść program pokazany na rysunku 10. w obszarze tworzenia programu i uruchom go. Wyjaśnij, na czym polega działanie tego programu. Odpowiedz na pytanie: *Jakie polecenia są realizowane, gdy naciśnięty jest klawisz z literą* d, *a jakie – gdy naciśnięty jest klawisz z literą* w*?* Dodaj jeszcze dwa polecenia warunkowe umożliwiające ruch duszka w pozostałych dwóch kierunkach. Dobierz samodzielnie klawisze (możesz wybrać dowolne wygodne dla ciebie klawisze). Zapisz program w pliku pod nazwą *rysowanie duszkiem.*

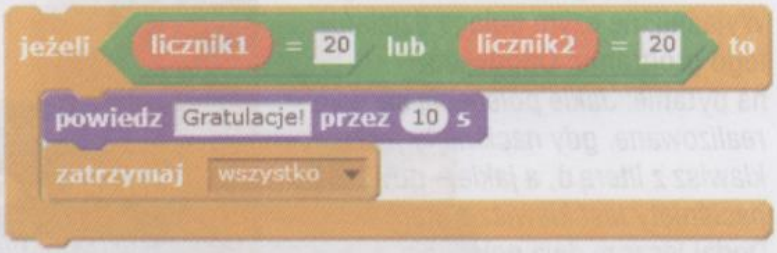
Rys. 10. *Skrypt – zadanie 1.*

2. Utwórz grę, w której wezmą udział dwa duszki. Jeden z duszków ma poruszać się losowo po całej scenie. Drugim duszkiem będziemy sterować (w lewo, w prawo, w górę, w dół), naciskając wybrane klawisze. Jeśli duszki się dotkną, otrzymujemy 10 punktów. Gra ma się zakończyć, gdy otrzymamy 100 punktów. Na koniec ma wyświetlić się komunikat „Brawo!" i gra ma zostać zakończona. Wybierz samodzielnie tło i postacie duszków (możesz również zmodyfikować istniejące postacie lub narysować nowe). Zapisz program w pliku pod nazwą określającą tematykę gry.
   Wskazówka: Własne sceny, tworzone np. w programie Paint, zapisz w plikach z rozszerzeniem *png*.
3. Otwórz plik zapisany w zadaniu 2. Dodaj trzeciego duszka, który ma poruszać się losowo po całej scenie. Jeśli nowy duszek i duszek sterowany klawiszami dotkną się, otrzymujemy -10 punktów. W skrypcie dla nowego duszka wprowadź drugi warunek zakończenia gry. Gra ma się zakończyć, gdy otrzymamy -50 punktów. Na koniec ma wyświetlić się komunikat „Nie udało się!" i gra ma zostać zakończona. Zapisz plik pod tą samą nazwą.
4. Utwórz grę dla dwóch graczy z udziałem trzech duszków. Jeden z duszków ma poruszać się losowo po całej scenie. Dwoma pozostałymi duszkami będą sterować gracze (w lewo, w prawo, w górę, w dół), naciskając wybrane

klawisze. Jeśli któryś z duszków sterowanych klawiszami dotknie się z duszkiem poruszanym losowo, gracz otrzymuje 2 punkty. Dla każdego z duszków utwórz oddzielny licznik, tworząc zmienne: *licznik1*, *licznik2*. Gra ma się zakończyć, gdy przynajmniej jeden z duszków uzyska 20 punktów. Na koniec ma wyświetlić się komunikat „Gratulacje!" i gra ma zostać zakończona. Wybierz samodzielnie tło i postacie duszków (możesz również zmodyfikować istniejące postacie lub narysować nowe). Zapisz program w pliku pod nazwą określającą tematykę gry.
Wskazówki: Na początku skryptu dla duszka losowo przemieszczającego się po scenie wyzeruj obydwie zmienne. Umieść też w tym skrypcie wszystkie polecenia **jeżeli**:

- dwa polecenia realizujące sytuację związaną z dotykaniem się duszków, w jednym zmieniamy o 2 punkty *licznik1*, a w drugim – o 2 punkty *licznik2*,
- trzecie polecenie, realizujące warunek zakończenia gry (rys. 11.), w którym jako warunku użyj elementu: 

.



***Rys. 11.** Realizacja warunku zakończenia gry – zadanie 4.*

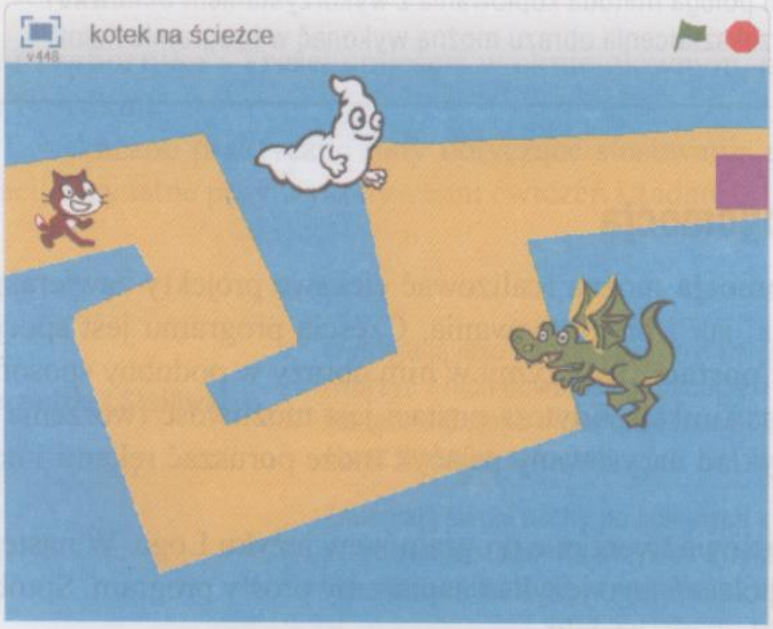
5. Utwórz grę dla dwóch graczy, w której będziemy sterować duszkami-pingwinami w lewo, w prawo, w górę i w dół, naciskając wybrane klawisze. Umieść na scenie duszka-gwiazdkę, który ma się losowo przemieszczać po całej scenie. Gdy duszek-pingwin dotyka duszka-gwiazdki – gwiazdka znika. Zdubluj duszki-gwiazdki, tak aby było ich na scenie piętnaście (rys. 12.). Dodaj zliczanie punktów dla każdego z duszków-pingwinów oddzielnie – po jednym punkcie za każdą dotkniętą gwiazdkę. Gra ma się zakończyć, gdy jeden z duszków uzyska co najmniej 15 punktów. Zapisz program w pliku pod nazwą *pingwiny i gwiazdki*.
Wskazówka: Aby ukryć duszka, należy użyć polecenia **ukryj** z grupy **Wygląd**. Na początku skryptu dla duszka-gwiazki umieść polecenie **pokaż**.

***Rys. 12.** Przykładowa scena – zadanie 5.*

6. Korzystając z możliwości zmiany tła, utwórz animowaną scenę pokazującą efekt zachodzącego słońca. Zapisz program w pliku pod nazwą *zachód słońca*.

Dla zainteresowanych

7. Narysuj ścieżkę dla duszka-kota, używając wybranych kolorów do pokolorowania ścieżki i tła (patrz wzór na rysunku 13.). Na końcu ścieżki umieść kwadrat w wybranym kolorze, symbolizujący prezent, do którego zmierza duszek-kot. Napisz grę, w której, używając klawiszy strzałek, będziesz przesuwał duszka-kota w stronę prezentu. Duszek-kot ma przesuwać się o 5 kroków tylko po ścieżce – nie może wyjść poza nią. Jeśli duszek dotknie tła, powinien cofnąć się o 5 kroków, czyli przesunąć się o -5 kroków. Po scenie mają losowo przelatywać dwa duszki: duszek-duch i duszek-smok. Gdy duszek-kot i duszek-smok dotkną się, duszek-kot wraca na pozycję początkową i licznik zmniejsza się o jeden. Gdy duszek-kot i duszek-duch dotkną się, licznik zwiększa się o jeden. Gracz wygrywa, gdy duszek-kot dojdzie do prezentu (czyli dotknie koloru, na jaki pomalowany jest prezent). Należy wtedy wyświetlić komunikat „Gratulacje" i uzyskaną liczbę punktów. Gracz przegrywa, gdy uzyska -5 punktów – w takim wypadku ma wyświetlić się komunikat „Nie udało się!". Zapisz program w pliku pod nazwą *kotek na ścieżce*.



Rys. 13. *Przykładowa scena – zadanie 7.*

8. Otwórz plik zapisany w ćwiczeniu 10. Zmodyfikuj grę, zwiększając prędkość lotu duszków na kolejnych poziomach, czyli zależnie od sumy uzyskanych punktów. Zapisz plik pod tą samą nazwą.
9. Utwórz grę według własnego pomysłu. Korzystając z **Pomocy**, zapoznaj się samodzielnie z możliwością wykorzystania w programie **stopera**. Uwzględnij wyświetlanie czasu trwania utworzonej gry. Zapisz program w pliku pod nazwą *moja gra*.