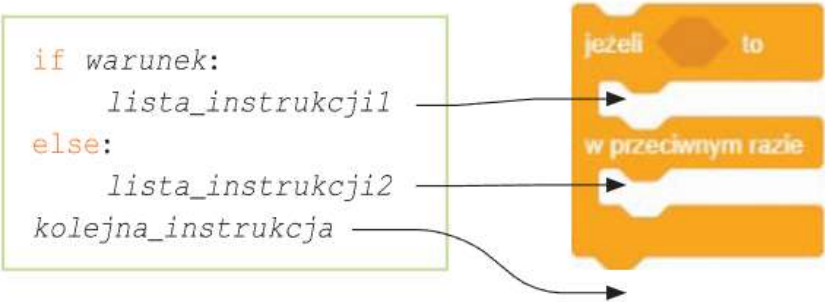


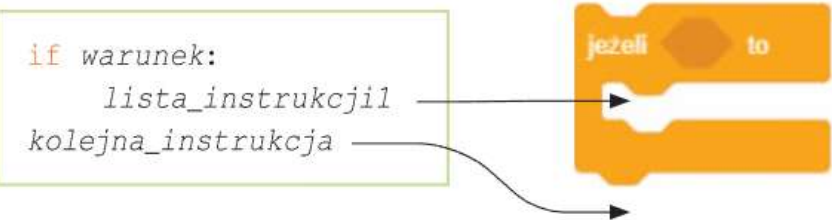
Zapisanie algorytmu z warunkami w języku Python

Aby w języku programowania napisać program realizujący algorytm z warunkami, stosujemy instrukcję warunkową.

W języku Python instrukcja warunkowa działa tak samo jak w języku Scratch i również występuje w wersjach pełnej i uproszczonej (rys. 5a, 5b). Jako *lista_instrukcji1* i *lista_instrukcji2* może wystąpić jedna instrukcja (w tym instrukcja warunkowa) lub więcej instrukcji. Po warunku i słowie kluczowym *else* umieszczamy dwukropek.



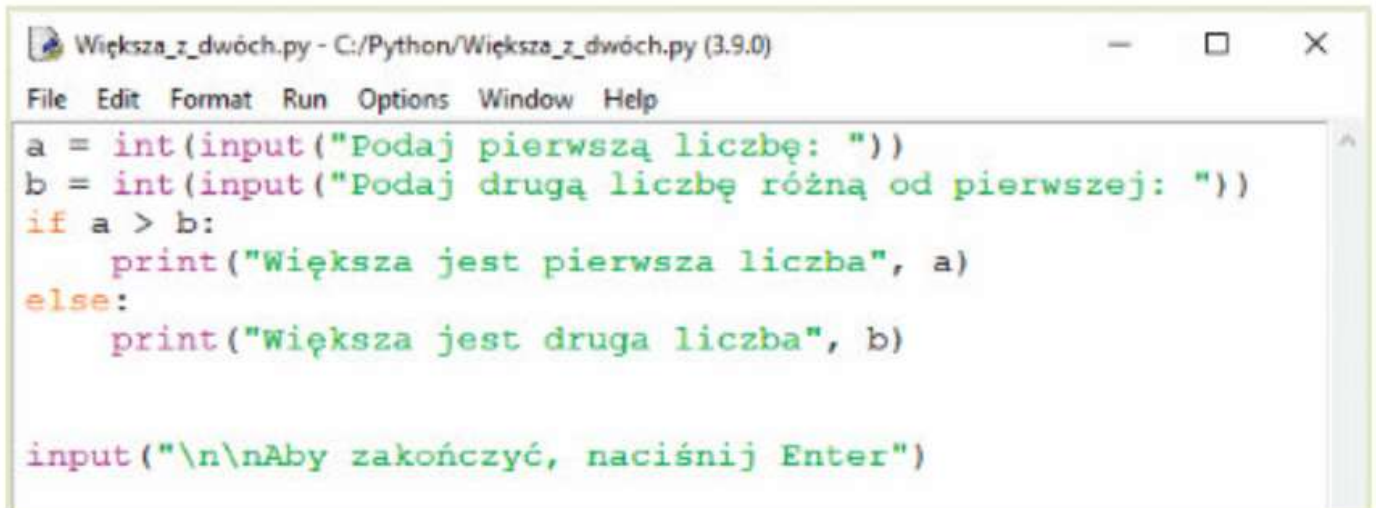
Rys. 5a. Ogólna postać instrukcji warunkowej w języku Python – po prawej odpowiadający jej element w języku Scratch



Operatory		Określenie	Przykład wyrażenia	Interpretacja wyrażenia
Porównania	==	równy	a == b	a równe b
	!=	różny	a != b	a różne od b
	<	mniejszy	a < b	a mniejsze od b
	>	wiekszy	a > b	a większe od b
	>=	wiekszy lub równy	a >= b	a większe lub równe b
	<=	mniejszy lub równy	a <= b	a mniejsze lub równe b
Logiczne	or	alternatywa logiczna (lub)	a < -5 or a > 5	a mniejsze od -5 lub a większe od 5
	and	koniunkcja logiczna (i)	a > 0 and a < 10	a większe od 0 i a mniejsze od 10
	not	negacja logiczna	not a == 5	a nierówne 5

Tabela 4. Podstawowe operatory porównania i logiczne w języku Python

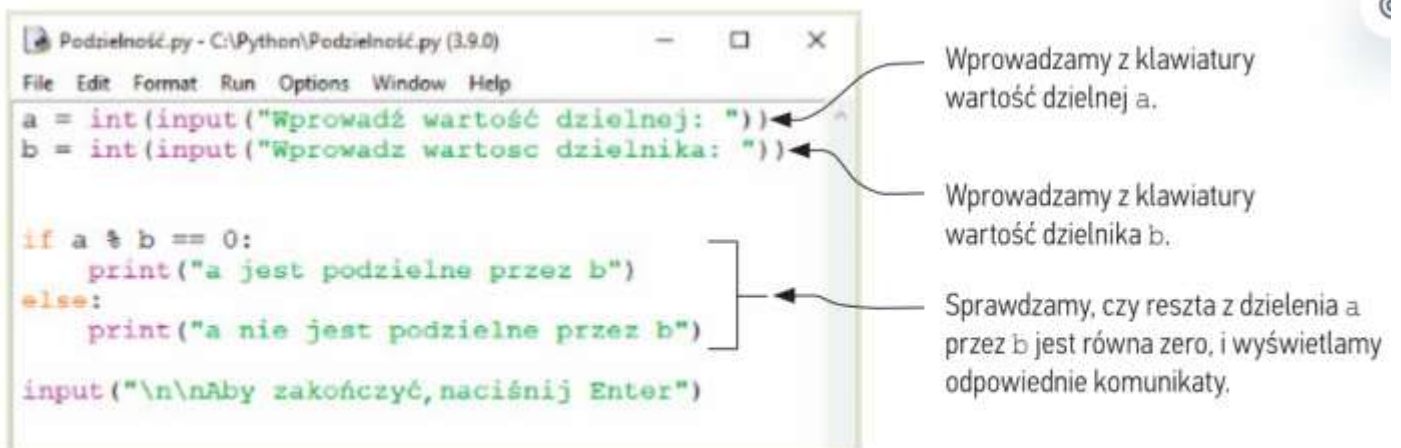
Zadanie 1: W programie IDLE utwórz nowy plik i wpisz poniższy przykład kodu programu. Przetestuj go na różnych liczbach.



```
Większa_z_dwóch.py - C:/Python/Większa_z_dwóch.py (3.9.0)
File Edit Format Run Options Window Help
a = int(input("Podaj pierwszą liczbę: "))
b = int(input("Podaj drugą liczbę różną od pierwszej: "))
if a > b:
    print("Większa jest pierwsza liczba", a)
else:
    print("Większa jest druga liczba", b)

input("\n\nAby zakończyć, naciśnij Enter")
```

Zadanie 2: Napisz program sprawdzający, czy wprowadzona z klawiatury dowolna liczba całkowita jest dodatnia. Jeśli jest dodatnia program powinien wyświetlić komunikat „LICZBA DODATNIA”, jeśli nie - „LICZBA NIEDODATNIA”. Program zapisz pod nazwą „Liczby” i uruchom.



```
Podzielność.py - C:/Python/Podzielność.py (3.9.0)
File Edit Format Run Options Window Help
a = int(input("Wprowadź wartość dzielnej: "))
b = int(input("Wprowadź wartość dzielnika: "))

if a % b == 0:
    print("a jest podzielne przez b")
else:
    print("a nie jest podzielne przez b")

input("\n\nAby zakończyć, naciśnij Enter")
```

Wprowadzamy z klawiatury wartość dzielnej a.

Wprowadzamy z klawiatury wartość dzielnika b.

Sprawdzamy, czy reszta z dzielenia a przez b jest równa zero, i wyświetlamy odpowiednie komunikaty.

Rys. 3. Program w języku Python realizujący algorytm badania podzielności liczb naturalnych

Zadanie 3. Przepisz powyższy program i zapisz w pliku „Podzielność”. Uruchom go kilkakrotnie testując na różnych liczbach.

Zadanie 4. Zmodyfikuj program zapisany w zadaniu 3, tak aby sprawdzał podzielność liczb tylko dla dzielnika b różnego od zera, a w przypadku gdy dzielnik b jest równy zero- wyświetlał komunikat „Wprowadzono dzielnik równy zero”. Zmodyfikowany program zapisz pod nazwą „Podzielnośćbezzera”

Wskazówka: wykorzystaj właściwe operatory (tabela 4.)

A tak to by wyglądało w Scratchu...

Zadanie 5. Przygotuj program z instrukcją warunkową z dowolnym operatorem logicznym.

