

Stosowanie instrukcji warunkowych w języku Python

Sytuacja warunkowa występuje wtedy, gdy wynik lub dalsze działanie zależą od spełnienia (lub niespełnienia) pewnego warunku.

Algorytm z warunkami (algorytm z rozgałęzieniami) - algorytm zawierający sytuacje warunkowe.

Przykład sytuacji warunkowej:

Czy pada deszcz?

Tak – zostaję w domu.

Nie – idę na spacer.

Czy jestem głodny?

Tak – zjem bułkę.

Nie – nie zjem bułki.

Arkusz kalkulacyjny Excel

fx					
=JEŻELI(A1>0;"Liczba dodatnia";"Liczba niedodatnia")					
C	D	E	F	G	H

Baltie



Scratch



C++

```
if(a > 0)
    cout << "Liczba dodatnia";
else
    cout << "Liczba niedodatnia";
```

Python

```
if a > 0:
    print("Liczba dodatnia")
else:
    print("Liczba niedodatnia")
```

Budowa najprostszej instrukcji warunkowej w języku Python:

```
if nasz_warunek:  
    Lista_instrukcji1  
else:  
    Lista_instrukcji2
```

Czyli:

Jeśli mamy warunek *nasz_warunek* i zostanie on spełniony to wykonaj *Lista_instrukcji1*. Jeśli nie zostanie on spełniony to zrób w przeciwnym razie *Lista_instrukcji2*.

Bardzo ważne jest zachowanie takiej budowy hierarchicznej (struktury) zapisu.

Decyduje ona o działaniu programu (komputer wie co ma robić w jakiej kolejności) oraz jego czytelności.

Wcięcia to zazwyczaj wciśnięty tab lub przyjęte 4 spacje.

Po warunku **if** oraz po **else** zawsze MUSI być dwukropek „:” (!).

Przykład prostego warunku logicznego:

$x > 0$, $a == 9$, liczba ≤ 10 , $a != 5$ (**większe** od 0, **równe** 9, ≤ 10 , **różne** od 5)

Przykład złożonego warunku logicznego:

$x > -5$ **and** $x \leq 0$, $x < -5$ **or** $x > 0$ ($x > -5$ **i musi być** ≤ 0 , $x < -5$ **lub może być** > 0)

Dla przejrzystości kodu ważne jest aby pomiędzy np. x i „=” lub „==” i cyfrą **9** były spacje.

Operatory porównania.

Operator	Wyjaśnienie
==	Równy
!=	Nie równy/różny
<>	Nie równy/różny
>, <	Większy/Mniejszy niż
>=, <=	Większy/Mniejszy niż lub równy

Operatory logiczne.

Operator	Określenie	Przykład warunku log.	Interpretacja wyrażenia
or	alternatywa logiczna (lub)	$a > 5$ or $a < -5$	a większe od 5 lub a mniejsze od 5
and	koniunkcja logiczna(i)	$a > 5$ and $a < 10$	a większe od 5 i a mniejsze od 10

Należy pamiętać, że:

operator „==” to operator porównania,

operator „=” to operator przypisania.

Przykład warunku logicznego prostego

Zadanie: Algorytm obliczania wartości bezwzględnej liczby całkowitej.

Dane: dowolna liczba całkowita: x.

Wynik: wartość bezwzględna liczby x równa w.

Kod:

```
x = int(input("Podaj liczbę x: "))

if x >= 0:
    w = x
else:
    w = -x
print("Wartość bezwzględna liczby x =", w)

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

Ćwiczenie 1.

1. Napisz specyfikację zadania i program realizujący algorytm sprawdzania, czy wprowadzona liczba jest dodatnia (≥ 0) czy niedodatnia (< 0).
2. Zapisz program w pliku pod nazwą **Liczby1**.

Ćwiczenie 2.

1. Napisz program, który zapyta użytkownika o wiek, a następnie sprawdzi i poinformuje go czy może w tym wieku grać w Wiedźmina. Użytkownik aby grać musi być pełnoletni.
2. Zapisz plik pod nazwą **Wiedzmin**.

Ćwiczenie 3.

1. Napisz specyfikację zadania i program realizujący algorytm sprawdzania, czy liczba jest podzielna przez 3. (Wskazówka: Oblicz resztę z dzielenia wprowadzonej liczby przez 3).
2. Zapisz program pod nazwą **Podzielna_przez_3**

Przykład warunku logicznego złożonego.

Alternatywa logiczna - **or** (lub).

Koniunkcja logiczna - **and** (i).

Ćwiczenie 4.

Klasa uczestniczy w lidze klas. Jeśli klasa w ostatnim semestrze uzyskała frekwencję powyżej 94% lub średnią ocen powyżej 4,0, to w nagrodę pojedzie na wycieczkę.

1. Napisz program sprawdzający, czy klasa zakwalifikuje się na wycieczkę. Frekwencję i średnią ocen wprowadzaj z klawiatury, a komunikat „nagroda” lub „brak nagrody” wyświetlaj na ekranie.
2. Zapisz program w pliku pod nazwą **Wycieczka**.

Ćwiczenie 5.

1. Napisz program sprawdzający czy po wpisaniu frekwencji oraz średniej ocen na koniec roku, uczeń otrzyma pasek czy nie.
2. Zapisz program pod nazwą **Pasek**

Ćwiczenie 6.

1. Napisz program, który odpowiada na pytanie, czy trzy podawane liczby całkowite są ustawione w porządku rosnącym.
2. Zapisz program pod nazwą **rosnace**

Ćwiczenie 7.

1. Napisz program, który dla zadanych dwóch liczb określi, czy pierwsza jest wielokrotnością drugiej.
2. Zapisz program pod nazwą **wielokrotnosc**