INSTRUKCJA ITERACYJNA WHILE

Do zapisywania algorytmów iteracyjnych, w których liczba iteracji nie jest z góry określona, możemy zastosować instrukcję **while**.

```
while warunek:
    lista_instrukcji
kolejna_instrukcja
```

Najpierw sprawdzany jest warunek. Jeżeli jest spełniony, to wykonywana jest `lista_instrukcji`.

Wewnątrz bloku `lista_instrukcji` powinna być zawsze umieszczona instrukcja, która zmienia wartość `warunku`. W przeciwnym wypadku pętla nigdy się nie zakończy.

`Lista_instrukcji` może zawierać jedną lub wiele instrukcji.

Instrukcje w pętli WHILE wykonują się aż do spełnienia określonego warunku.

Przykład 1.

Napisz program wyświetlający rosnąco co 1 od wartości równej 1 do wartości 10.

```
i = 0
while i < 10:
    i = i + 1 # i += 1
    print(i)
```

```
i = 1
while i < 11:
    print(i)
    i = i + 1 # i += 1
```

Przykład 2.

Zadanie: Napisz program obliczający sumę wpłat aż do osiągnięcia lub przekroczenia założonej kwoty.

Dane: liczba rzeczywista dodatnia (**kwota**), oznaczająca kwotę potrzebną na zakup telewizora, ciąg dowolnych liczb rzeczywistych dodatnich (**wplata**), oznaczających kolejne comiesięczne wpłaty.

Wynik: liczba rzeczywista dodatnia (**suma_wplat**), oznaczająca wartość sumy wszystkich wpłat.

```
kwota = float(input("Podaj potrzebną kwotę: "))
suma_wplat = 0

while suma_wplat < kwota:
    wplata = float(input("Wprowadź wpłatę: "))
    suma_wplat = suma_wplat + wplata

print("Suma wpłat wynosi:", suma_wplat, "zł")

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

Ćwiczenie 1.

1. Przepisz powyższy program. Zapisz pod nazwą **zbiorka**.
2. Uruchom i przetestuj dla różnych wartości zmiennych.

Ćwiczenie 2.

1. Zmodyfikuj program zbiorka tak, aby na ekranie wyświetlały się również komunikaty z bieżącą suma, uzyskaną po kolejnej wpłacie, a na koniec różnica pomiędzy kwota wpłaconą a założoną (*nadpłata*).
2. Zapisz plik pod nazwą **zbiorka_szczegoly**

Ćwiczenie 3.

1. Otwórz program **zbiorka_szczegoly** i dodaj do niego sprawdzanie poprawności sprowadzonych danych *kwota* i *wplata* – zgodnie ze specyfikacją zadania:
   - ✓ Jeżeli użytkownik wprowadzi niepoprawną wartość zmiennej *kwota*, wyświetl komunikat: „Wprowadzono błędną liczbę” i zakończ program.
   - ✓ Jeżeli użytkownik wprowadzi niepoprawną wartość zmiennej *wplata*, wyświetl komunikat: „Wprowadzono błędną liczbę” i zignoruj wprowadzoną wartość.
2. Zapisz plik pod nazwą **zbiorka_szczegoly_korekta**.

Ćwiczenie 4.

1. Napisz specyfikację zadania i program, który będzie obliczał sumę liczb całkowitych wprowadzanych z klawiatury, aż do wprowadzenia zera, którego wpisanie kończy zliczanie. Wynik sumowania wyprowadź na ekran.
2. Zapisz plik pod nazwą **suma_while**

Podpowiedź: Na początku programu przypisz zmiennej liczba wartość różną od zera, np. 1, a zmiennej suma: 0.