

Temat lekcji:

Generowanie ciągu Fibonacciego

Cele lekcji:

- Uczniowie rozumieją, czym jest ciąg Fibonacciego.
- Uczniowie poznają funkcje generatora w Pythonie.
- Uczniowie potrafią analizować i modyfikować kod generujący ciąg Fibonacciego.

Czas trwania: 45 minut

Wprowadzenie (10 minut)

Co to jest ciąg Fibonacciego?

- Przedstawienie definicji: każdy wyraz (poza pierwszymi dwoma) jest sumą dwóch poprzednich.
- Przykład: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Pytania do uczniów:

- Czy znacie jakieś zastosowania ciągu Fibonacciego?
- Czy potraficie podać kolejne liczby w ciągu?

Analiza kodu (15 minut)

Kod:

python

```
def genFib(n):  
    a, b = 0, 1  
    for i in range(n+1):  
        yield a  
        a, b = b, a + b  
  
print(list(genFib(9)))
```

Omówienie krok po kroku:

- `def genFib(n):`: definicja funkcji generatora.
- `a, b = 0, 1`: inicjalizacja pierwszych dwóch liczb.
- `for i in range(n+1):`: pętla wykonująca się $n+1$ razy.
- `yield a`: zwraca bieżącą wartość `a` bez kończenia funkcji.
- `a, b = b, a + b`: aktualizacja wartości `a` i `b`.

Wynik działania:

- `list(genFib(9))` → [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]

Ćwiczenie praktyczne (5 minut)

Zmień kod tak, aby generował 100 liczb ciągu Fibonacciego.

Podsumowanie i refleksja (10 minut)

Pytania do uczniów:

- Co daje użycie `yield` zamiast `return`?
- Czy znasz zastosowanie ciągu Fibonacciego w sztuce?