

Diskreetti matematiikka Tehtävä 8

Rekursio

Tehtävä 41.

Toteuta materiaalissa ollut kertoman laskeva koodi

a) rekursiivisena

```
# Description: Calculate the factorial of a number  
# Author: Aman Mughal
```

```
# Function to calculate the factorial of a number  
def calculate_factorial(n: int) -> int:  
# Description: Calculate the factorial of a number  
# Author: Aman Mughal
```

```
# Function to calculate the factorial of a number recursively  
def calculate_factorial_recursive(n: int) -> int:  
    # Check if the number is negative or zero  
    # If it is negative, return None. if it is zero, return 1  
    # Otherwise, return the number multiplied by the factorial of the number subtracted by 1
```

```
    if n == 0:  
        return 1  
    elif n < 0:  
        return None  
    else:  
        return n * calculate_factorial_recursive(n - 1)
```

```
number = int(input("Enter a number to calculate its factorial: "))  
result = calculate_factorial_recursive(number)
```

```
if result is None:  
    print("Factorial of negative numbers is not defined.")  
else:  
    print(f"The factorial of {number} is {result}.")
```

b) ei-rekursiivisena

```
# Description: Calculate the factorial of a number  
# Author: Aman Mughal
```

```
def calculate_factorial_non_recursive(n: int) -> int:  
    # Check if the number is negative or zero  
    # If it is negative, return None. if it is zero, return 1
```

```

        #Otherwise calculate the factorial of the number using a for loop and return the result

    if n == 0:
        return 1
    elif n < 0:
        return None
    else:
        factorial = 1
        for i in range(1, n + 1):
            factorial *= i
        return factorial

number = int(input("Enter a number to calculate its factorial: "))
result = calculate_factorial_non_recursive(number)

if result is None:
    print("Factorial of negative numbers is not defined.")
else:
    print(f"The factorial of {number} is {result}.")

```

Tehtävä 42.

Toinen klassinen rekursiotehtävä on Fibonaccin lukusarja, eli 1, 1, 2, 3, 5, 8, 13, ... Tee tästä rekursiivinen sovellus, jolla voi laskea järjestysnumeroltaan haluamansa luvun. Esimerkiksi luku 13 on 7:s luku.

```

# Description: Python program to display the Fibonacci sequence
# Author: Aman Mughal

```

```

def calculate_fibonacci_recursive(n: int) -> int:
    if n <= 1:
        return n
    else:
        return calculate_fibonacci_recursive(n - 1) + calculate_fibonacci_recursive(n - 2)

nterms = int(input("Enter the number of terms: "))
result = calculate_fibonacci_recursive(nterms)

print(f"The Fibonacci number of the terms {nterms} is {result}.")

print("The Fibonacci sequence is: ")
for i in range(nterms + 1):
    print(calculate_fibonacci_recursive(i))

```

Tehtävä 43.

Talletat määräaikaistilille 10 000 euroa kymmeneksi vuodeksi. Vuosikorkoa saat talletuksellesi 1,3%. Tee ohjelma, joka laskee paljonko tililläsi on rahaa 10 vuoden kuluttua. Tee ohjelmasta kohtuullisen yleiskäyttöinen ja toteuta ohjelma

a) rekursiivisena

```
# Description: Python program to calculate the interest for your savings  
# Author: Aman Mughal
```

```
# Function to calculate the the balance after a certain number of years with a given interest rate
```

```
def calculate_balance_after_years_recursive(balance: float, interest_rate: float, years: int):  
    # Check if the number of years is 0  
    # If it is, return the balance  
    # Otherwise, calculate the balance after one year and call the function again with the new balance  
  
    if years == 0:  
        return balance  
    else:  
        balance = balance * (1 + interest_rate)  
        return calculate_balance_after_years_recursive(balance, interest_rate, years - 1)
```

```
balance_str = input("Enter The starting balance e.g 1000 usd: ")  
balance = float(balance_str.split(" ")[0])  
currency = balance_str.split(" ")[1]  
interest_rate = float(input("Enter the interest rate %: ")) / 100  
years = int(input("Enter the number of years: "))
```

```
result = calculate_balance_after_years_recursive(balance, interest_rate, years)
```

```
print(f"The balance after {years} years is {result} {currency}.")
```

b) ei-rekursiivisena

```
# Description: Python program to calculate the interest for your savings  
# Author: Aman Mughal
```

```
def calculate_balance_after_years_non_recursive(balance: float, interest_rate: float, years: int):  
    # Calculate the balance after a certain number of years with a given interest rate  
    balance = balance * (1 + interest_rate) ** years  
    return balance
```

```
balance_str = input("Enter The starting balance e.g 1000 usd: ")  
balance = float(balance_str.split(" ")[0])  
currency = balance_str.split(" ")[1]  
interest_rate = float(input("Enter the interest rate %: ")) / 100
```

```

years = int(input("Enter the number of years: "))

result = calculate_balance_after_years_non_recursive(balance, interest_rate, years)

print(f"The balance after {years} years is {result} {currency}.")

```

Tehtävä 44.

Tarinan mukaan hollantilainen Peter Minuit osti Manhattanin saaren paikallisilta intiaaneilta vuonna 1626 helyillä, joiden arvo on n. 24 dollaria. Jos intiaanit olisivat tallettaneet tämän helyistä saadun rahasumman paikalliseen pankkiin ja saaneet siitä korkoa vaikkapa saman verran, kuin edellisessä tehtävässä, niin paljonko intiaaneilla olisi ollut rahaa tilillä.

- a) 10 vuoden kuluttua Eli tässä voidaan käyttää edellisen tehtävän laskenta logiikkaa, mutta sijoitus korko ja vuodet vaihtuvat.
 - 10 vuoden kuluttua intiaaneilla olisi ollut 27.31 dollaria.
- b) vuonna 1800
 - Vuonna 1800 intiaaneilla olisi ollut 227.11 dollaria.
- c) tänään
 - Tänään intiaaneilla olisi ollut 4047.26 dollaria.
- d) Entä jos korko nousee prosenttiyksikön, paljonko intiaanit olisivat silloin netonneet?

```

def calculate_balance_after_years_non_recursive(balance: float, interest_rate: float, years: int)
    # Calculate the balance after a certain number of years with a given interest rate
    balance = balance * (1 + interest_rate) ** years
    return balance

balance = 24
currency = "usd"
interest_rate_1 = 0.013
interest_rate_2 = 0.023

years = 10

result_1 = calculate_balance_after_years_non_recursive(balance, interest_rate_1, years)
result_2 = calculate_balance_after_years_non_recursive(balance, interest_rate_2, years)

net_gain = result_2 - result_1

print(f"Interest rate increased by 1%, the net gain is {net_gain} {currency}.")

```

- Nettovoitto olisi 2.81 dollaria.

Aman Mughal 20/03/2023