Informatik 1 - Biomedical Engineering

Tutor Session 2 - Branching

Overview

- The if-Statment
- For Loop
- While Loop
- Examples

The if-Statement

If expression_1 is true, then instruction 1 should happen. If expression_2 is true, instruction 2 should happen. If both are not true, the else instruction will be executed.

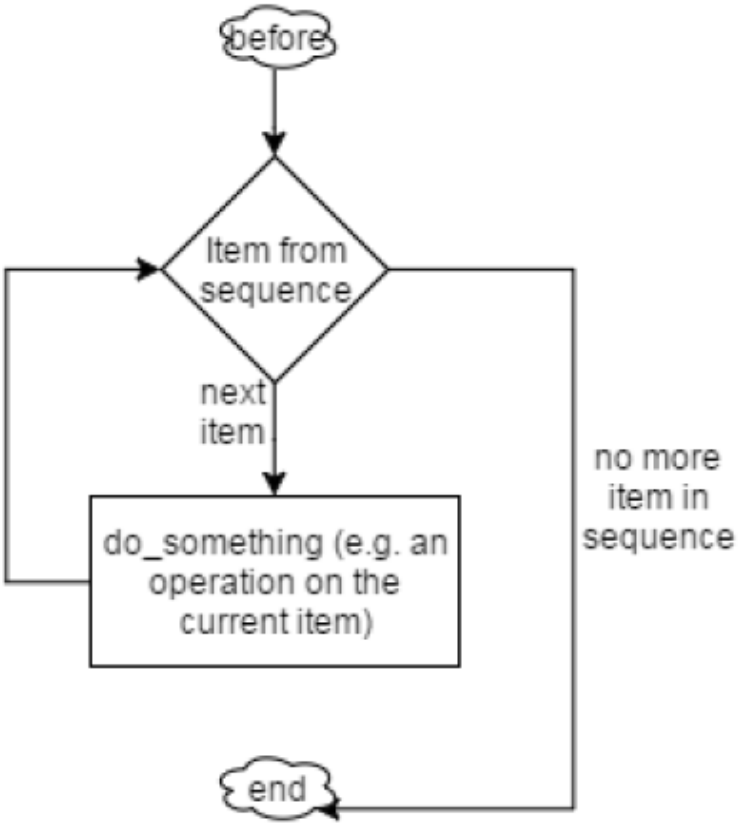
- Examples:

In [ ]:
```
if expression_1:
    #instruction 1
elif expression_2: #optional
    #instruction 2
else: #optional
    #else instruction
```

In [ ]:
```
weather = input("How is the weather today (rainy/sunny): ")

if weather == "rainy":
    print("clean your room!")
elif weather == "sunny":
    print("you can go swimming :-)")
else:
    print("pff. don\'t have a recommendation.")
print("anyhow, watch a movie at night.")
```

For Loop



In [ ]:
```
for item in sequence:
    statement(s)
```

In [ ]:
```
#summarize even numbers
res = 0
for number in range(11):
    if number % 2 == 0:
        res += number
print(res)
```

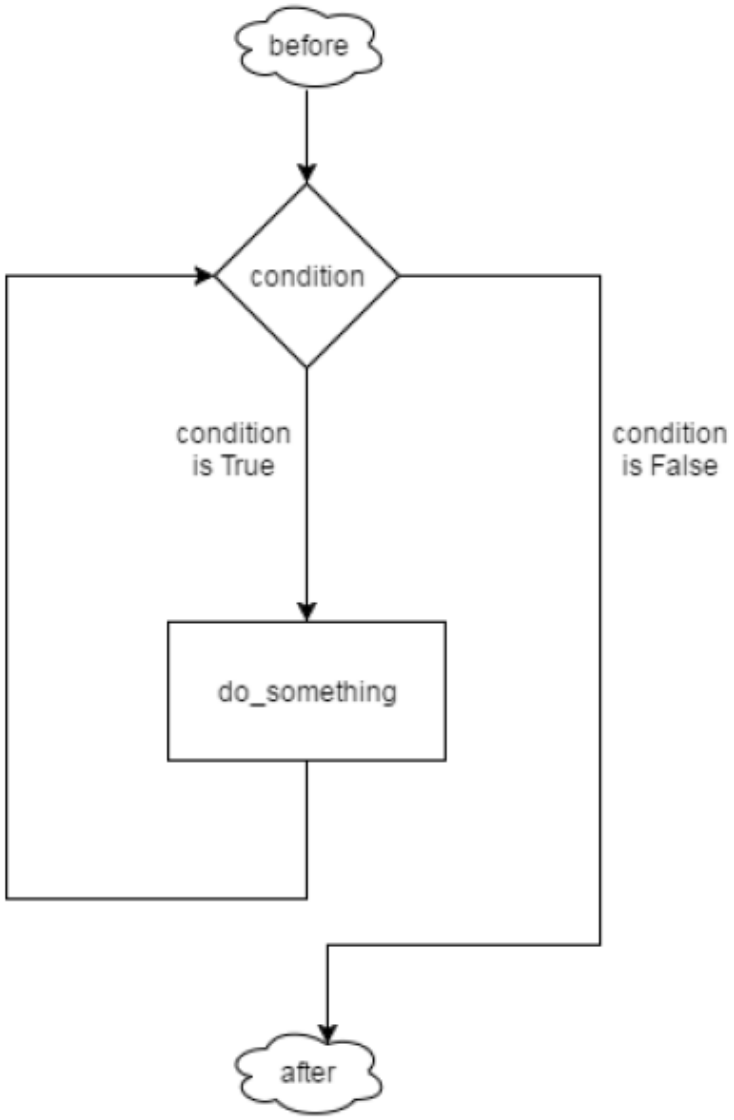
Using range to iterate:

In [ ]:
```
for i in range(0,9):
    print(i)
```

In [ ]:
```
for i in range(9):
    print(i)
```

In [ ]:
```
for i in range(5,9):
    print(i)
```

While Loop



In [ ]:
```
while condition:
    #do_something
```

In [ ]:
```
secure_pwd = "123"
user_inp = ""
attempts = 0

while user_inp != secure_pwd:
    user_inp = input("Enter pwd: ")
    attempts += 1
print("Authenticated after ", attempts," attempts")
```

Example Program

Calculating π using Leibniz's formula.

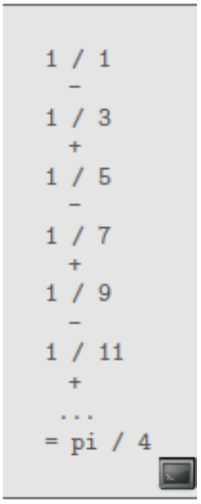
$$\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = \frac{\pi}{4}$$

written as a series:

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \ldots = \frac{\pi}{4}$$

In [ ]:
```
#possible solution
ITERATIONS = 1000
subtotal = 0.0
for n in range(ITERATIONS):
    subtotal += (-1)**n / (2*n + 1)
pi = subtotal * 4
print("Pi is appr.:", pi)
```

Now try and print out the series: Write a program that prints the first six elements of the Leibniz series to the console. (See picture)



Hint: Use a While Loop (or For Loop) which runs from 1 to 11

In [ ]:
```
#possible solution:
last_denominator = 11
operator = "-"
act = 1
while act <= last_denominator:
    print(1, "/", act)
    print(" ", operator)
    act += 2
    if operator == "-":
        operator = "+"
    else:
        operator = "-"
print(" ...\n= pi / 4")
```