Coding For Creative Robotics

Week 4: Repetition without being repetitive?

Sonia Litwin
25/10/2024

Agenda

Lecture (30’):
- Last week recap
-Real world loops in:
- Robotics
- Nature
- Biomedicine

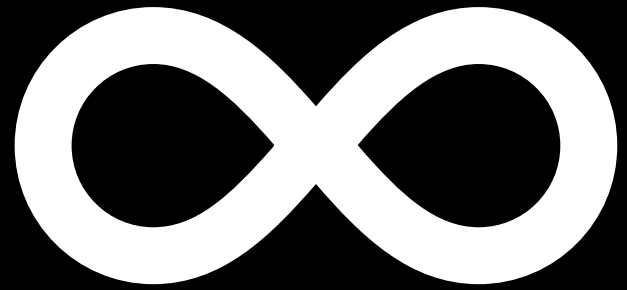
Tutorial (120’):
- Homework questions
- Loops: guided exercises
- Loops: practice

Loops

while, for

Iteration and repetition in Python and beyond

Sonia Litwin
25/10/2024



Loops, Repetition, Feedback

Where can we find loops in the real world?

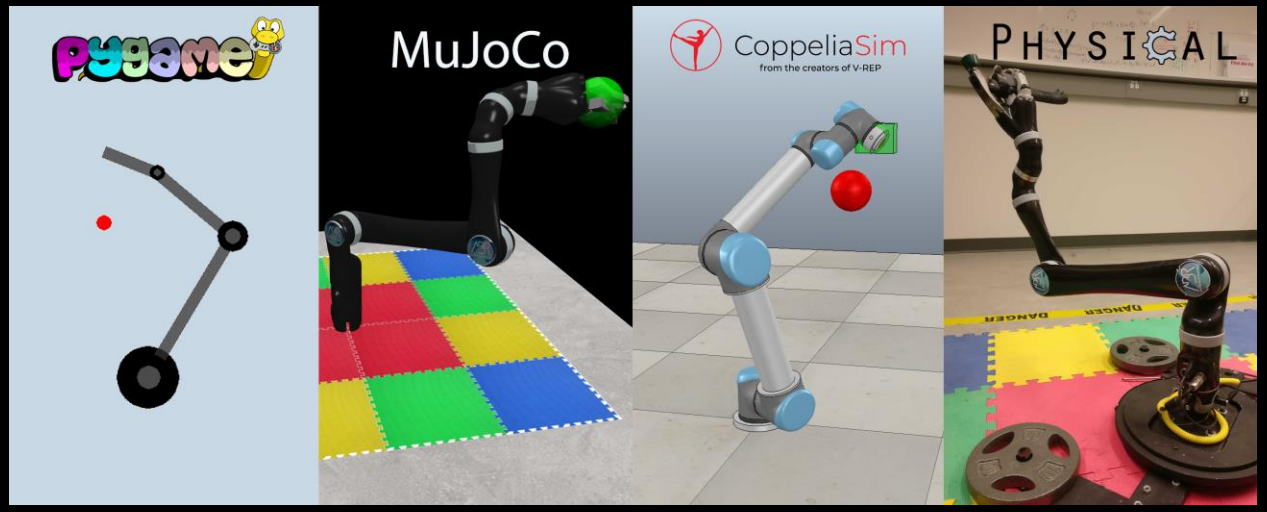
Iterative Processes

Robotics & Automation

Robotic arms

Industrial robotic arms are an essential part of modern factory automation, where precision and repetitive tasks are crucial. Programming an industrial arm involves defining the sequence of movements it must perform to complete tasks such as welding, assembly, or painting.

G-code, a language used to instruct CNC machines, often employs loops to repeat tool paths, optimizing both time and code simplicity.



Iterative Processes

Robotics & Automation

G-Code

Robotic arms in manufacturing are programmed to perform precise and repetitive tasks. G-code is a widely used language that instructs machines on how to move.

```
; Begin G-code program
G90         ; Set to absolute positioning
G21         ; Set units to millimeters

; Define a square path
G1 X0 Y0 F1500   ; Move to starting point at speed 1500 mm/min

; Loop to draw the square 4 times
M98 P100 L4      ; Call subroutine 100, repeat 4 times

M30         ; End of program

; Subroutine to draw a square
O100
G1 X50 Y0       ; Move to X=50
G1 X50 Y50      ; Move to Y=50
G1 X0 Y50       ; Move to X=0
G1 X0 Y0        ; Return to starting point
M99           ; End of subroutine
```

Iterative Processes

Robotics & Automation

Navigation

In robotics, proximity sensors are a fundamental tool for navigation. These sensors allow a robot to detect obstacles and navigate around them effectively. The data from the proximity sensor is continuously looped through, allowing the robot to make real-time decisions about its path.

Time (ms)	Distance (cm)	Status
0	150	Safe
100	148	Safe
200	120	Safe
300	85	Warning
400	60	Warning
500	45	Critical
600	30	Critical
700	25	Critical
800	80	Warning
900	120	Safe

Iterative Processes

Robotics & Automation

Voice control

When implementing voice control in robotics, loops play a crucial role in maintaining constant communication between the human and the robot. The robot needs to continuously listen for voice commands in what's called a "listening loop", while simultaneously running a separate loop to process and respond to these commands.

"Start recording new command." ...this is operator's command (italic text)...

"I'm recording" ...the system says (bold text)...

"Search black disks"

"I'm searching ... Four disks were found"

"Move on first"

"I'm moving ... Done"

"Take it."

"Ok"

"Move on position alpha."

"I'm moving ... Done"

"Put it"

"Ok"

"Stop recording."

"I stop the recording. Please, say new command"

"Search" "Disks" "Done"

"New command is entered and named: Search disks. Is it right?"

"Yes"

...now, the newly defined command may be used...

"Repeat command"

"Enter command"

"Search disks"

"OK"

...now the system repeats the command until no more disks are found...

"No object found. Repeating done."

...now all the disks on the scene are transported into position alpha...

Voice control loop: The robot finds the remaining three disks and puts them into the selected area. If no disk is found the robot interrupts the execution of the given command and waits for a new command.

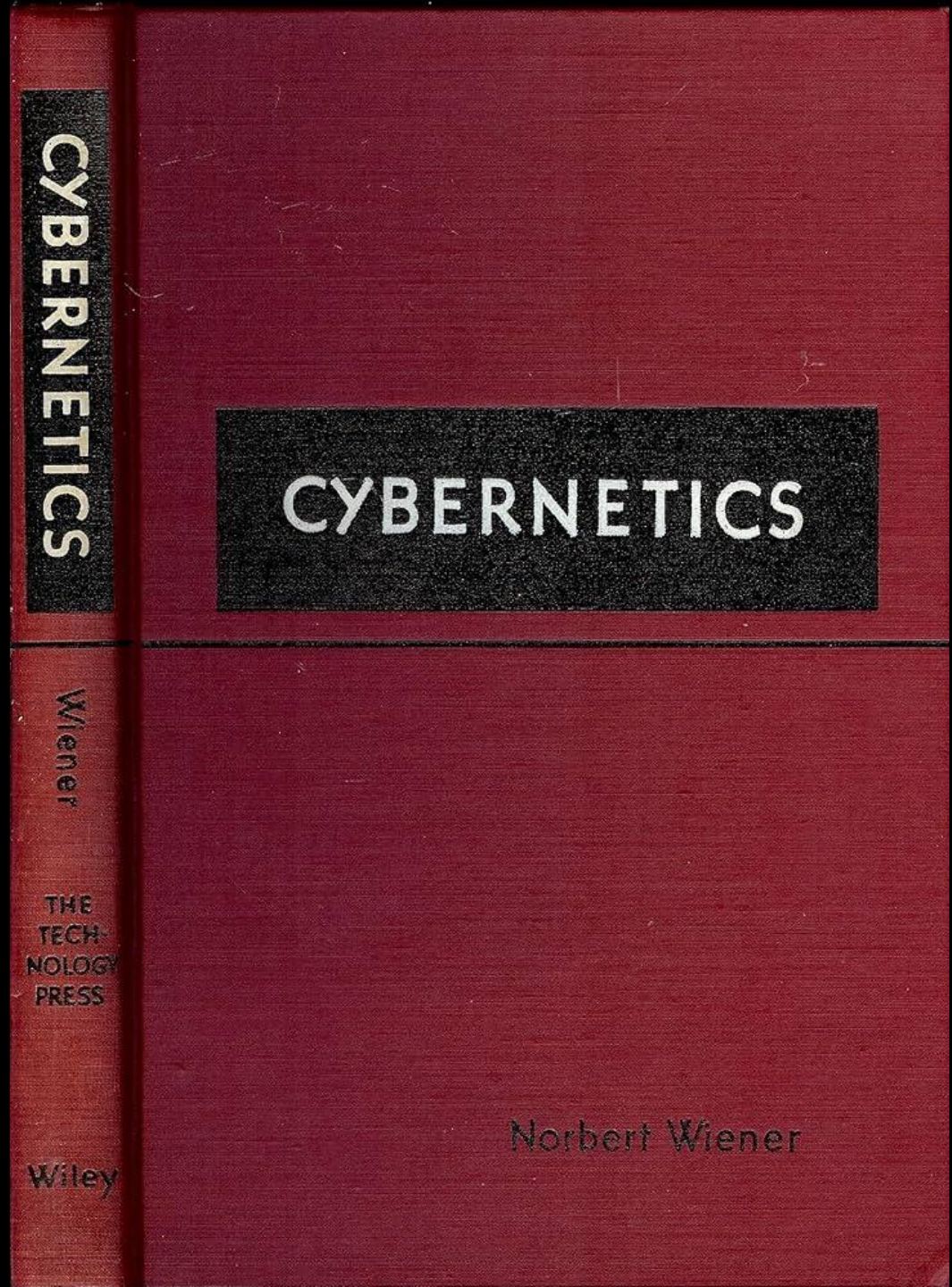
Source: Holada, Miroslav & Pelc, Martin. (2008). The Robot Voice-control System with Interactive Learning. 10.5772/6284.

Iterative Processes

Cybernetics

Control and Communication in the Animal and the Machine

Cybernetics is an interdisciplinary field that studies regulatory systems, their structures, constraints, and possibilities. Coined by Norbert Wiener in 1948, it focuses on how systems use feedback to maintain stability and achieve desired outcomes. Feedback loops are essential in cybernetics—they allow a system to adjust its behaviour based on the difference between the desired and actual outputs.

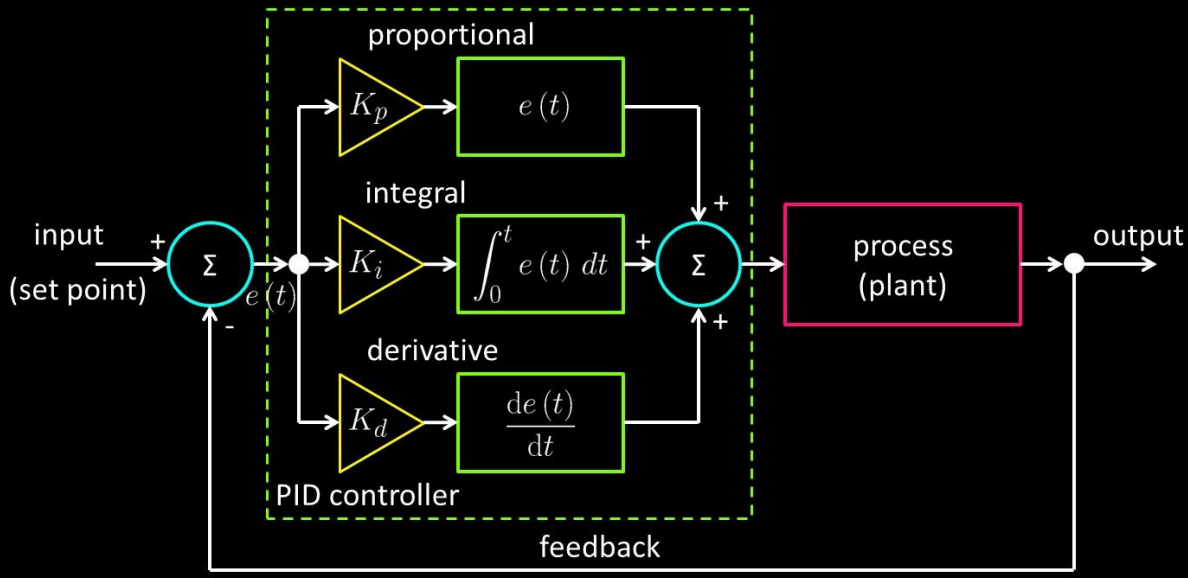


Iterative Processes

PID

Building on the principles of cybernetics, PID controllers are a practical application of feedback loops in engineering and robotics.

They continuously calculate an error value as the difference between a desired setpoint and a measured process variable, applying corrections based on proportional, integral, and derivative terms. This results in precise control of mechanisms, such as maintaining the speed of a motor or the position of a robotic arm.



Source:
https://www.digikey.com/en/maker/projects/introduction-to-pid-controllers/763a6dca352b4f2ba00adde46445ddeb

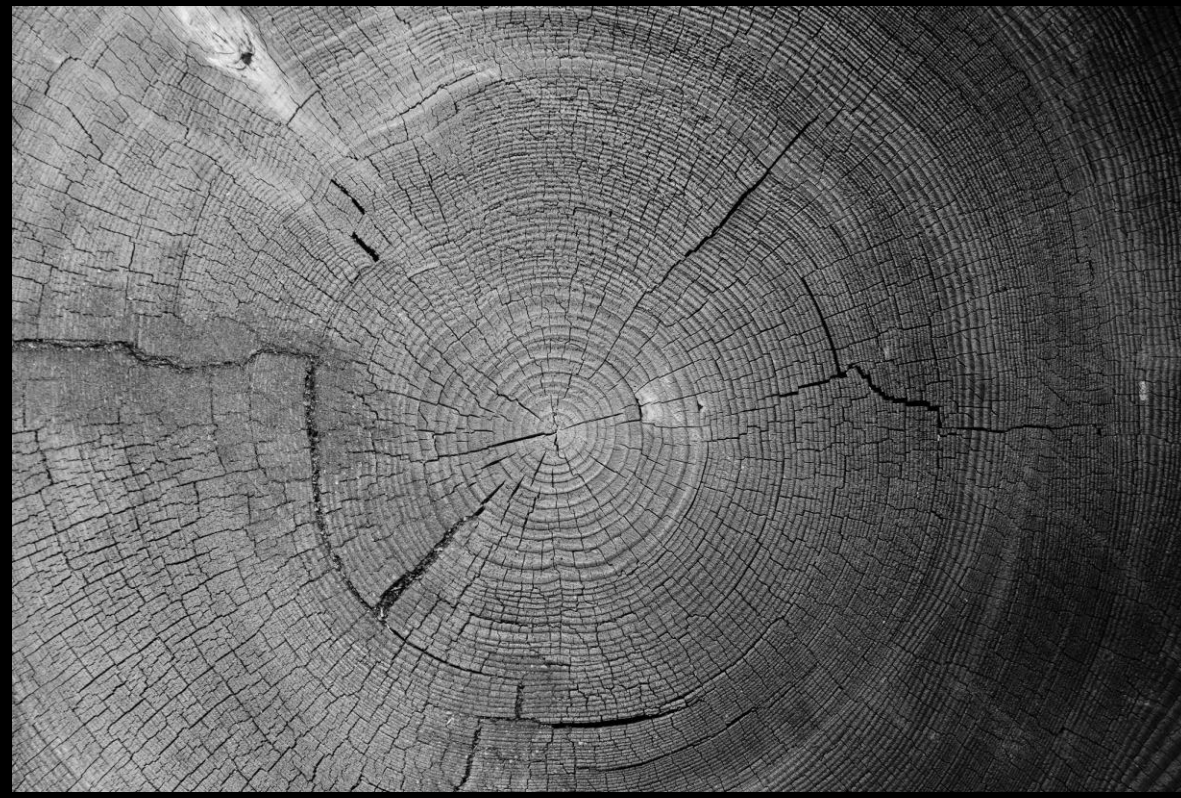
Iterative Processes

Nature

Fibonacci, Fractals

```
# Generate the first 10 numbers of the Fibonacci sequence

fib_sequence = [0, 1]
for i in range(2, 10):
    next_number = fib_sequence[i-1] + fib_sequence[i-2]
    fib_sequence.append(next_number)
print(fib_sequence)
```



Iterative Processes

Art & Design

Repetition principle

Repetition is not only crucial in nature and programming but also serves as a foundational principle in art and design. By repeating elements such as shapes, colors, or motifs, artists and designers create rhythm and harmony in their work. This repetition can lead to a sense of unity and cohesiveness, guiding the viewer's eye through the artwork.

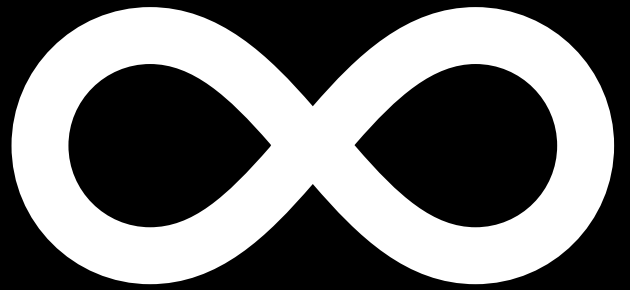
Iterative Processes

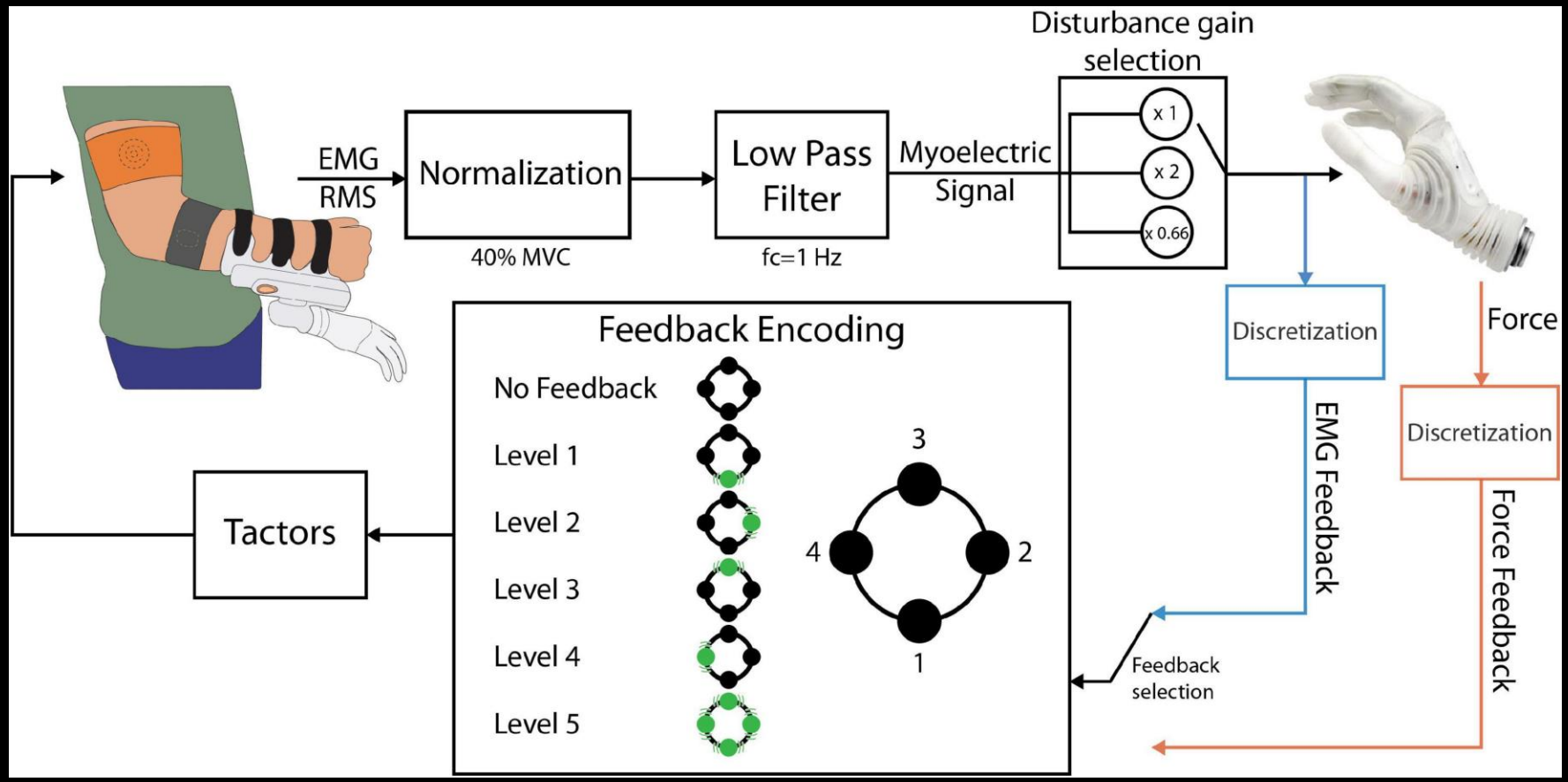
Biomedicine

EMG Prosthetics

In the field of healthcare, loops are essential for real-time prosthetics control.

EMG-based prosthetic arm uses loops to continuously read muscle signals, interpret them, and move accordingly. The prosthetic arm receives signals from the user's muscles, which are processed in a loop to determine how the arm should move.





EMG Prosthetics: The closed-loop prosthesis control scheme. The ME signal was sent to the prosthesis as a closing velocity or opening command.

Source: Tchimino, J., Dideriksen, J. L., & Dosen, S. (2022). EMG feedback outperforms force feedback in the presence of prosthesis control disturbance. Frontiers in Neuroscience, 16, Article 952288. https://doi.org/10.3389/fnins.2022.952288

Iterative Processes

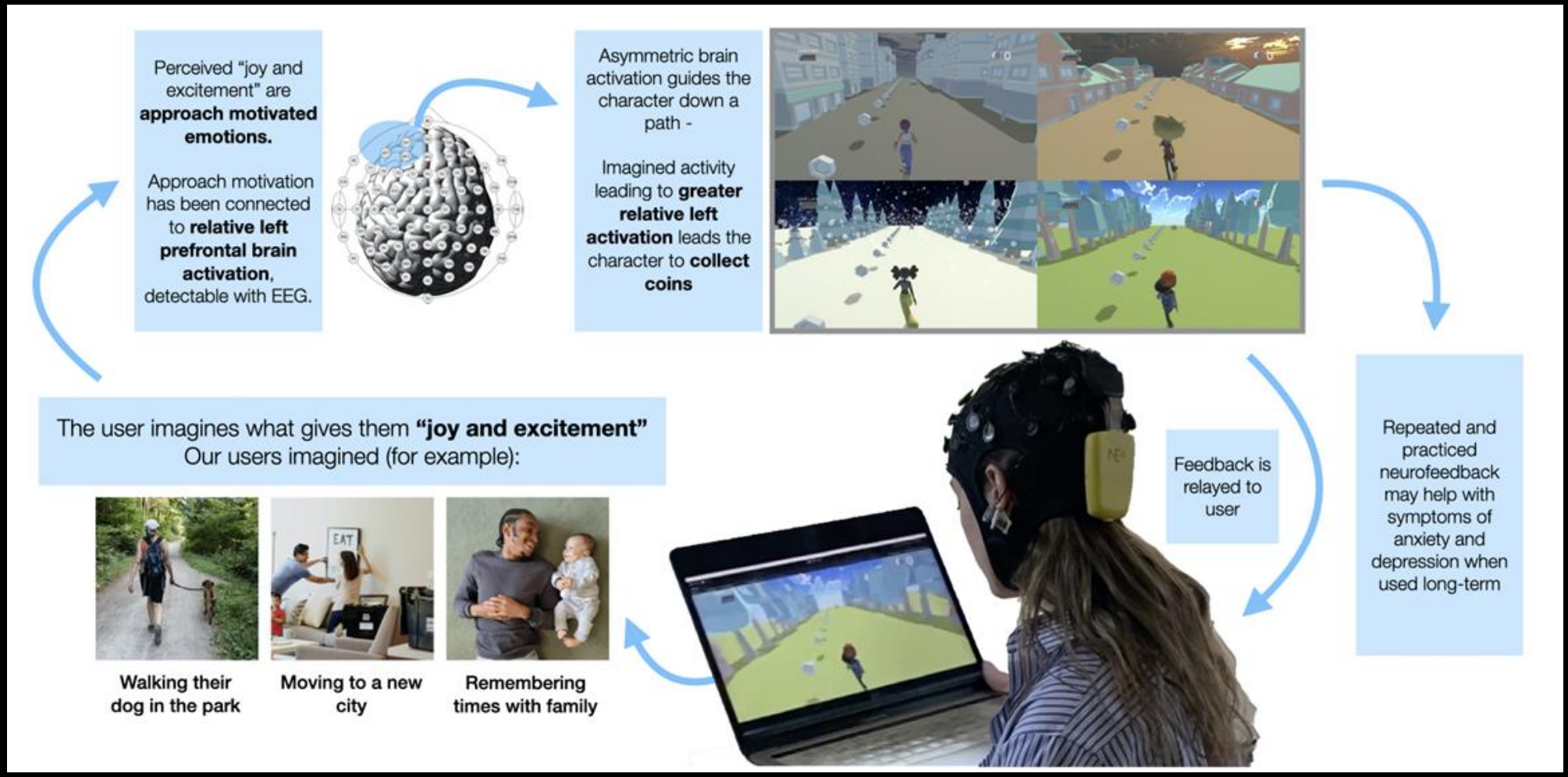
Biomedicine

Biofeedback

Biofeedback is a powerful application of loops in the healthcare.

Muse is a headband that monitors brain activity using EEG sensors and provides real-time feedback to help users manage stress, improve focus, or achieve meditative states.

Muse biofeedback.
Source: https://choosemuse.com/



EEG Biofeedback: Joie: the player learns mental strategies for joy and excitement. Brain asymmetries are mapped to an endless runner game. The user's game character collects coins when users activate relative frontal left asymmetry tied to "joy and excitement."

Source: Vujic, A., Nisal, S., & Maes, P. (2023). Joie: A joy-based brain-computer interface (BCI). In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23) (pp. 1-14). Association for Computing Machinery. https://doi.org/10.1145/3586183.360676

while

Python loop 1

The while loop is a control flow statement that allows code to be executed repeatedly based on a given Boolean condition. The code block inside the loop is executed as long as the condition evaluates to True. In our example, the loop will print the value of count and increment it by 1 each time until count is no longer less than 5.

```
count = 0
while count < 5:
    print(count)
    count += 1
```

count = count + 1

Infinite while

Python loop 1

When we didn’t intent to create a program that runs forever

```
while True:
    print("Running indefinitely...")
```

Ctrl + C

Intentionally infinite

Waiting for a command…

While loops are commonly used in scenarios where the loop needs to continue until a certain condition is met. For example, you might use a while loop to keep asking the user for input until they provide a valid response.

for

Python loop 2

A for loop is used for iterating over a sequence, such as a list, tuple, dictionary, set, or string. In the example, we're iterating over each character in the string 'Python' and printing it.

```
for letter in 'Python’:
    print( f'Current Letter: {letter}’ )
```

The range() function is a built-in function that generates a sequence of numbers. It's commonly used in for loops when you need to execute a loop a specific number of times. In the example, range(1, 6) generates numbers from 1 to 5, and the loop prints each iteration number. You can also include a step parameter to increment by values other than 1.

```
for i in range(1, 6):
    print(f'Iteration number: {i}')
```

continue, break, return

Control statements

Control statements alter the flow of execution within loops. The break statement terminates the loop entirely and proceeds to the next statement after the loop. The continue statement skips the rest of the current iteration and moves to the next iteration of the loop. The return statement is used within functions to exit the function and optionally pass back a value to the caller.

- **break:** Exits the loop immediately.
- **continue:** Skips to the next iteration.
- **return:** Exits a function, optionally returning a value.

```
for num in range(10):
    if num == 5:
        print('Breaking out of the loop’)
        break
    print(f'Current number: {num}')
```

Nested loops

Repetition inside repetition

Nested loops are loops that exist within another loop. The inner loop completes all its iterations for every single iteration of the outer loop. In the example, the outer loop runs from 1 to 3, and the inner loop runs from 1 to 2. This results in a total of 6 iterations, combining each value of i with each value of j. Nested loops are particularly useful when working with matrices or multi-dimensional arrays but can lead to increased complexity and execution time, so they should be used judiciously.

```
for i in range(1, 4):
    for j in range(1, 3):
        print(f'i = {i}, j = {j}')
```

Adding chaos

The use of randomness

Noise + iteration = art?

TouchDesigner

Tutorial

Loops

Sonia Litwin
25/10/2024