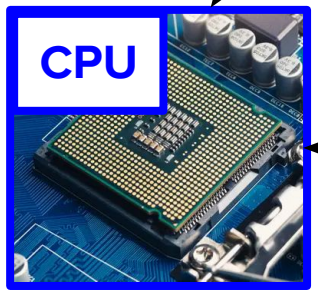
NumPy and Efficiency
Day 8 – PH 365

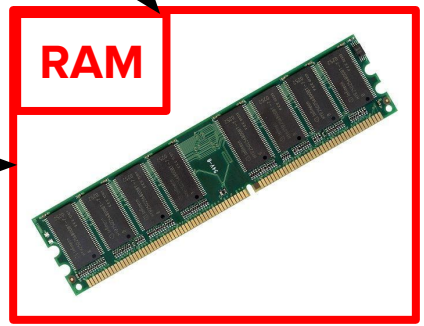
21 Oct 2024

```
import numpy as np
```

Stored libraries

NumPy library moved from storage to RAM for immediate access



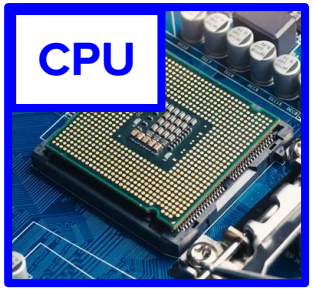


Variable created called **np**

Imported libraries

The Old Way: Using Loops

```
L = {a list}
operate = {a numerical function}

M = []
for i in range(len(L)):
    M.append(operate(L[i]))
```

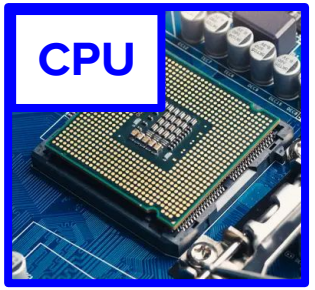


Computer processor:

1. Setup L, operate
2. Create empty list M
3. Do operation on first element of L
4. Append result to M
5. Do operation on second element of L
6. Append result to M
7. Do operation on third element of L
8. Append result to M
9. Repeat many more times...

The New Way: Using NumPy “Vectorization”

```
A = {an array}
operate = {a numerical function}

B = operate(A)
```

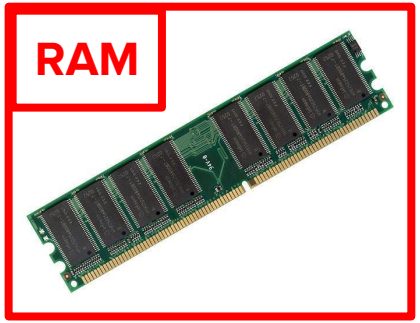


Computer processor:

1. Setup A, operate
2. In parallel, apply operate to every element of A
3. Store result in B

The Old Way: Using Loops

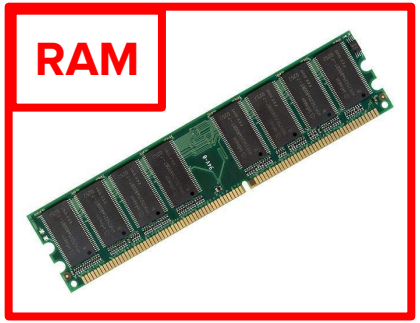
```
L = {a list}
operate = {a numerical function}

M = []
for i in range(len(L)):
    M.append(operate(L[i]))
```



RAM:

1. Store list (L) and function (operate)
2. Store new list (M)
3. After first append, if more space is needed, store M into a new area of memory with additional space
4. After second append, if more space is needed, store M into a new area of memory with additional space
5. After third append, if more space is needed...
6. Repeat many more times...

The New Way: Using NumPy “Vectorization”




```
A = {an array}
operate = {a numerical function}

B = operate(A)
```

RAM:

1. Store array (A) and function (operate)
2. Store array (B)

Syntax and Structure

```
L = {a list}
A = {an array}
operate = {a numerical function}
```

```
M = []
for i in range(len(L)):
    M.append(operate(L[i]))
```

```
B = operate(A)
```

Comparing Efficiency Quantitatively

```
import time

t0 = time.time()
```

This asks the computer for the exact time, and stores it in **`t0`**

Can be used multiple times like a stopwatch to time how long chunks of code take to run