

Prototype of Optimization in High-Performance Computing

Nitish Dhinakaran

University of the Cumberland

MSCS 532 M20 – Algorithms and Data Structures

Dr. Brandon Bass

August 09, 2025

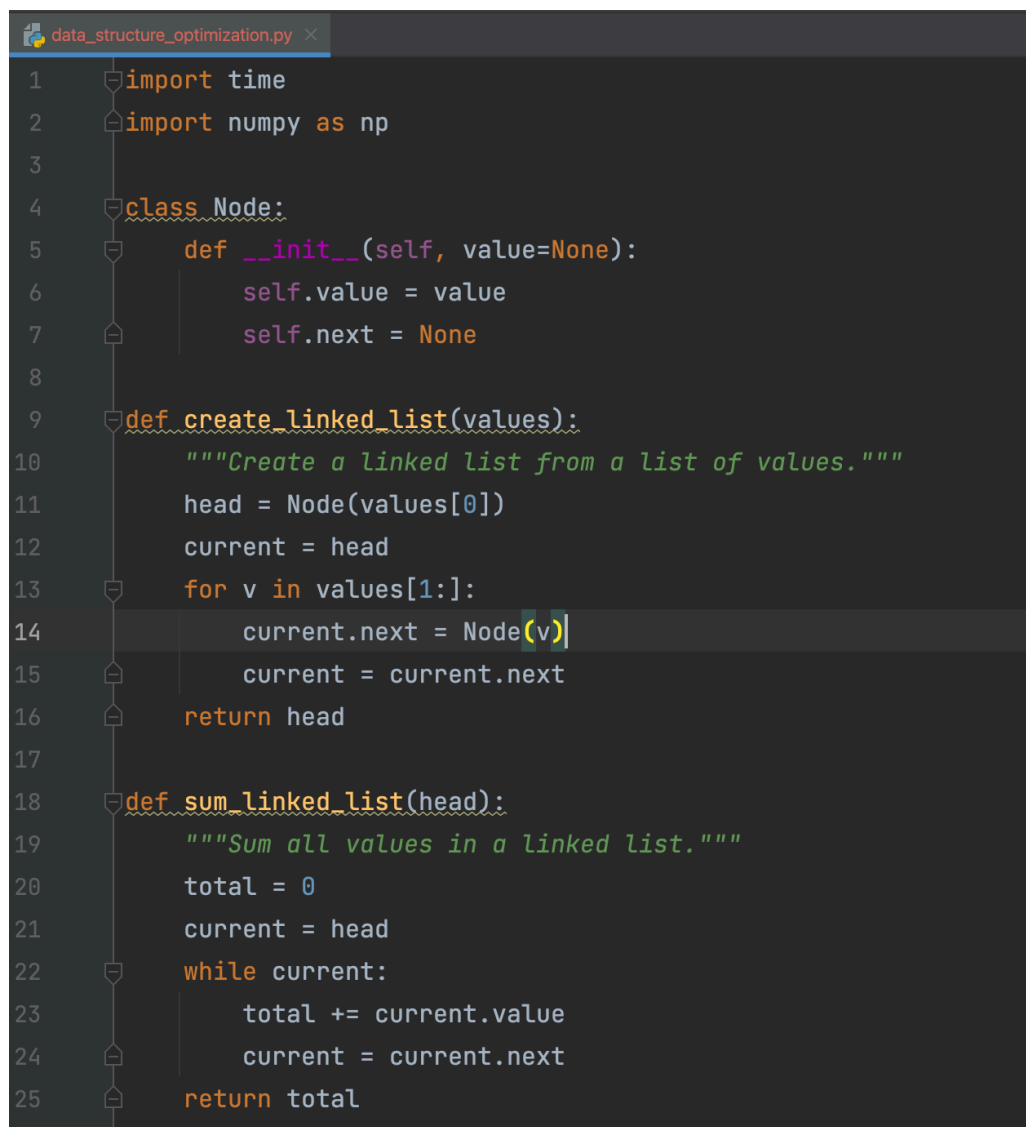
Python Prototype depicting Optimization in High-Performance Computing

Github Link of Source code

https://github.com/ndhinaharan36295/MSCS-532_Final-Project

Screenshot of Source code

Linked list structure:-

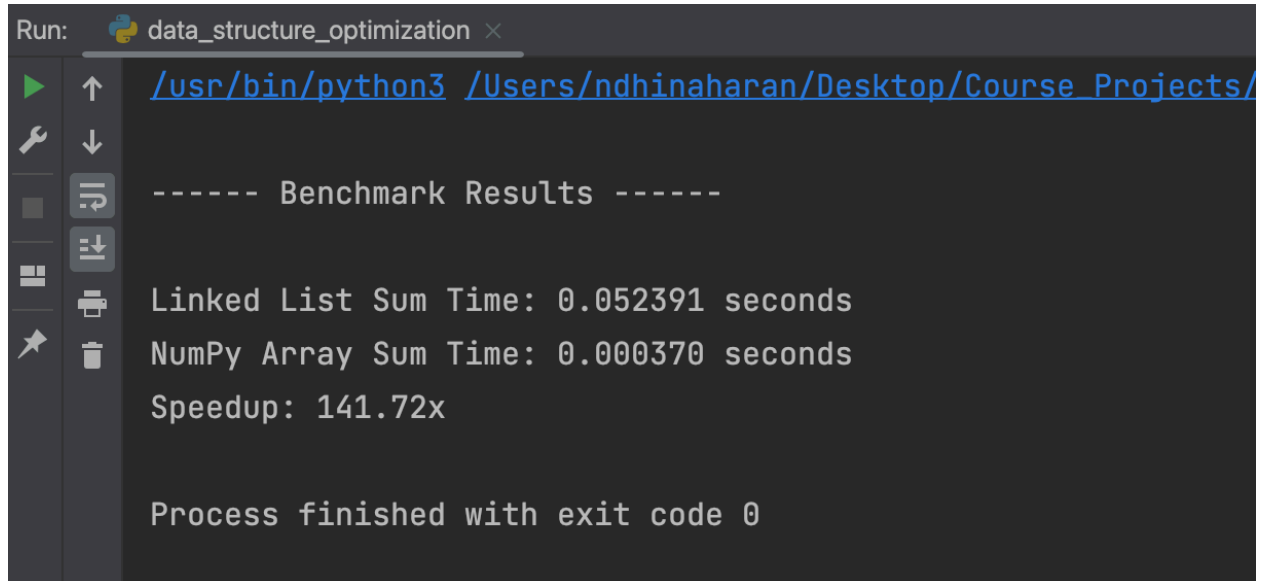
A screenshot of a code editor window titled 'data_structure_optimization.py'. The code implements a linked list structure with three functions: Node class, create_linked_list, and sum_linked_list. The code is as follows:

```
1 import time
2 import numpy as np
3
4 class Node:
5     def __init__(self, value=None):
6         self.value = value
7         self.next = None
8
9 def create_linked_list(values):
10     """Create a linked list from a list of values."""
11     head = Node(values[0])
12     current = head
13     for v in values[1:]:
14         current.next = Node(v)
15         current = current.next
16     return head
17
18 def sum_linked_list(head):
19     """Sum all values in a linked list."""
20     total = 0
21     current = head
22     while current:
23         total += current.value
24         current = current.next
25     return total
```

Benchmark implementation to compare Linked list and NumPy array:-

```
data_structure_optimization.py x
26
27 def benchmark(N=1_000_000):
28     """Benchmark linked list vs NumPy array summation."""
29     data = list(range(N))
30
31     # Linked list benchmark
32     linked_head = create_linked_list(data)
33     start = time.perf_counter()
34     sum_linked_list(linked_head)
35     linked_time = time.perf_counter() - start
36
37     # NumPy array benchmark
38     arr = np.array(data, dtype=np.int32)
39     start = time.perf_counter()
40     np.sum(arr)
41     numpy_time = time.perf_counter() - start
42
43     print("\n----- Benchmark Results ----- \n")
44     print(f"Linked List Sum Time: {linked_time:.6f} seconds")
45     print(f"NumPy Array Sum Time: {numpy_time:.6f} seconds")
46     print(f"Speedup: {linked_time / numpy_time:.2f}x")
47
48 if __name__ == "__main__":
49     benchmark()
```

Screenshot of Benchmarking results to show optimization



```
Run: data_structure_optimization x
/usr/bin/python3 /Users/ndhinaharan/Desktop/Course_Projects/

----- Benchmark Results -----

Linked List Sum Time: 0.052391 seconds
NumPy Array Sum Time: 0.000370 seconds
Speedup: 141.72x

Process finished with exit code 0
```

The screenshot shows a terminal window with a dark background. The title bar at the top reads "Run: data_structure_optimization x". The command prompt shows the path `/usr/bin/python3 /Users/ndhinaharan/Desktop/Course_Projects/`. The output displays benchmark results for a linked list and a NumPy array, showing a significant speedup of 141.72x for the linked list operation. The process finished with exit code 0.

From the above screenshot, it can be seen that the performance improves massively.