111. Mathematical analysis of Non-Recursive Algorithms

AIM: To find the analysis of Non-Recursive algorithm

PROGRAM:

```python
import time
import random

def find_max(arr):
    """ Function to find the maximum element in an array """
    if not arr:
        return None  # Handle empty array case

    max_element = arr[0]  # Initialize max_element with the first element

    for i in range(1, len(arr)):
        if arr[i] > max_element:
            max_element = arr[i]

    return max_element
array_size = 1000
random_array = [random.randint(1, 1000) for _ in range(array_size)]

start_time = time.time()
result = find_max(random_array)
end_time = time.time()
execution_time = end_time - start_time
print(f"Maximum element found: {result}")
print(f"Execution time: {execution_time} seconds")
```

INPUT:
```
array_size = 1000
```

OUTPUT:

```
Maximum element found: 999
Execution time: 4.124641418457031e-05 seconds
```

TIME COMPLEXITY: O(n)