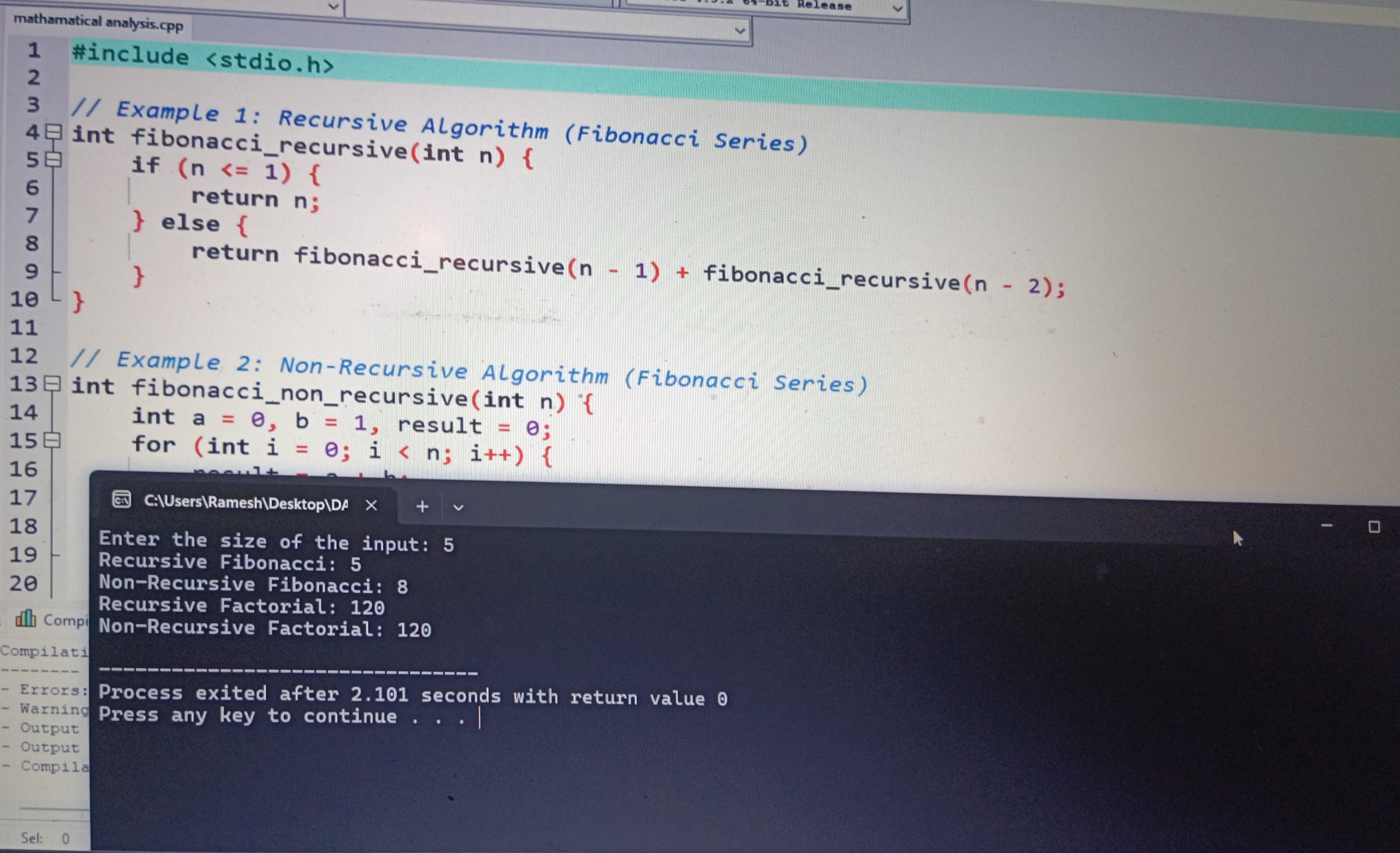
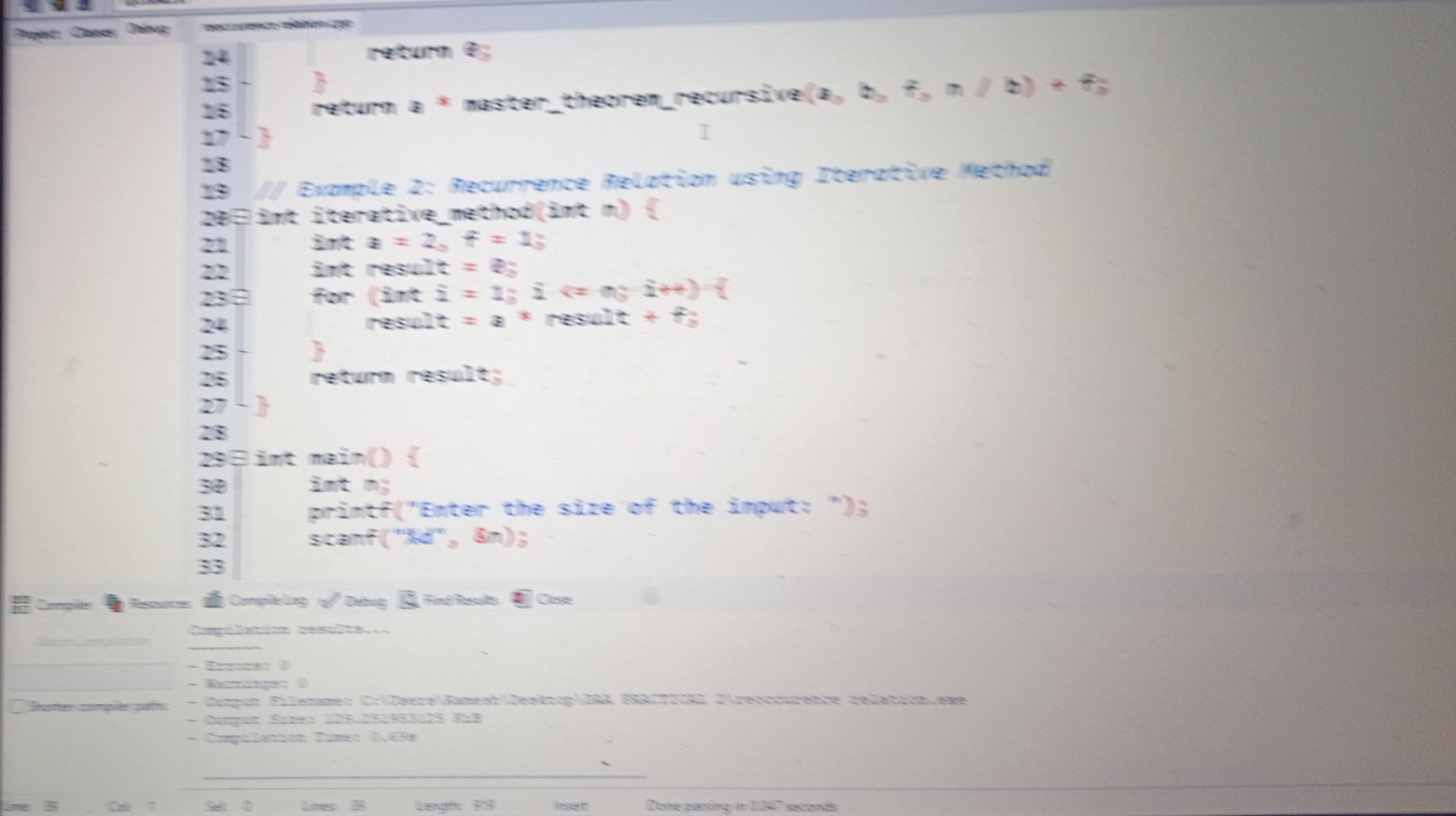


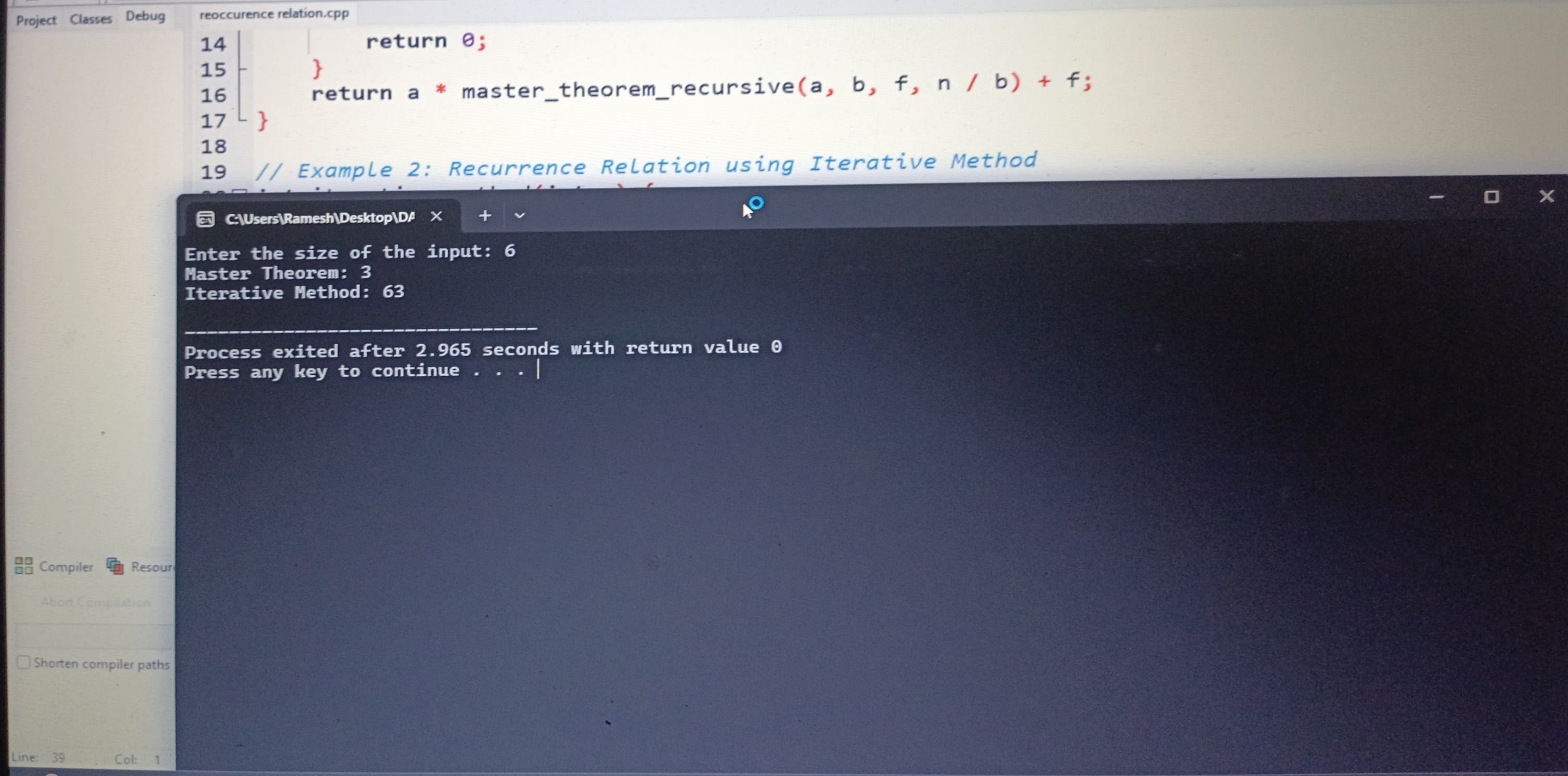
```c
#include <stdio.h>

// Example 1: Recursive Algorithm (Fibonacci Series)
int fibonacci_recursive(int n) {
    if (n <= 1) {
        return n;
    } else {
        return fibonacci_recursive(n - 1) + fibonacci_recursive(n - 2);
    }
}

// Example 2: Non-Recursive Algorithm (Fibonacci Series)
int fibonacci_non_recursive(int n) {
    int a = 0, b = 1, result = 0;
    for (int i = 0; i < n; i++) {
        result = a + b;
```

```
Enter the size of the input: 5
Recursive Fibonacci: 5
Non-Recursive Fibonacci: 8
Recursive Factorial: 120
Non-Recursive Factorial: 120

--------------------------------
Process exited after 2.101 seconds with return value 0
Press any key to continue . . .
```



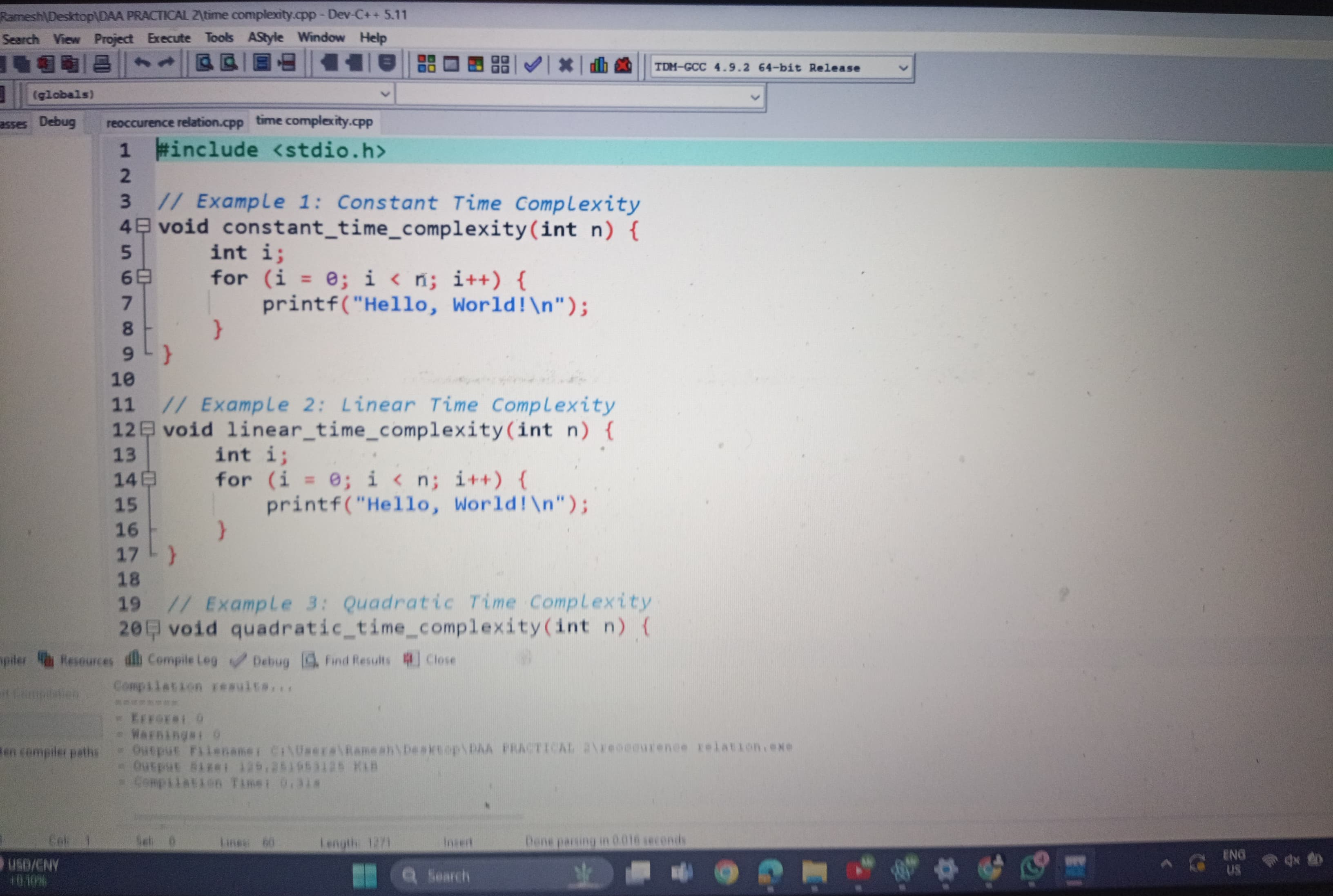
```c
        return 0;
    }
    return a * master_theorem_recursive(a, b, f, n / b) + f;
}

// Example 2: Recurrence Relation using Iterative Method
int iterative_method(int n) {
    int a = 2, f = 1;
    int result = 0;
    for (int i = 1; i <= n; i++) {
        result = a * result + f;
    }
    return result;
}

int main() {
    int n;
    printf("Enter the size of the input: ");
    scanf("%d", &n);

```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\recourence relation.exe
- Output Size: 129.251953125 KiB
- Compilation Time: 1.69s
```



```
            return 0;
        }
        return a * master_theorem_recursive(a, b, f, n / b) + f;
}

// Example 2: Recurrence Relation using Iterative Method
```

```
Enter the size of the input: 6
Master Theorem: 3
Iterative Method: 63

--------------------------------
Process exited after 2.965 seconds with return value 0
Press any key to continue . . .
```



Ramesh\Desktop\DAA PRACTICAL 2\time complexity.cpp - Dev-C++ 5.11

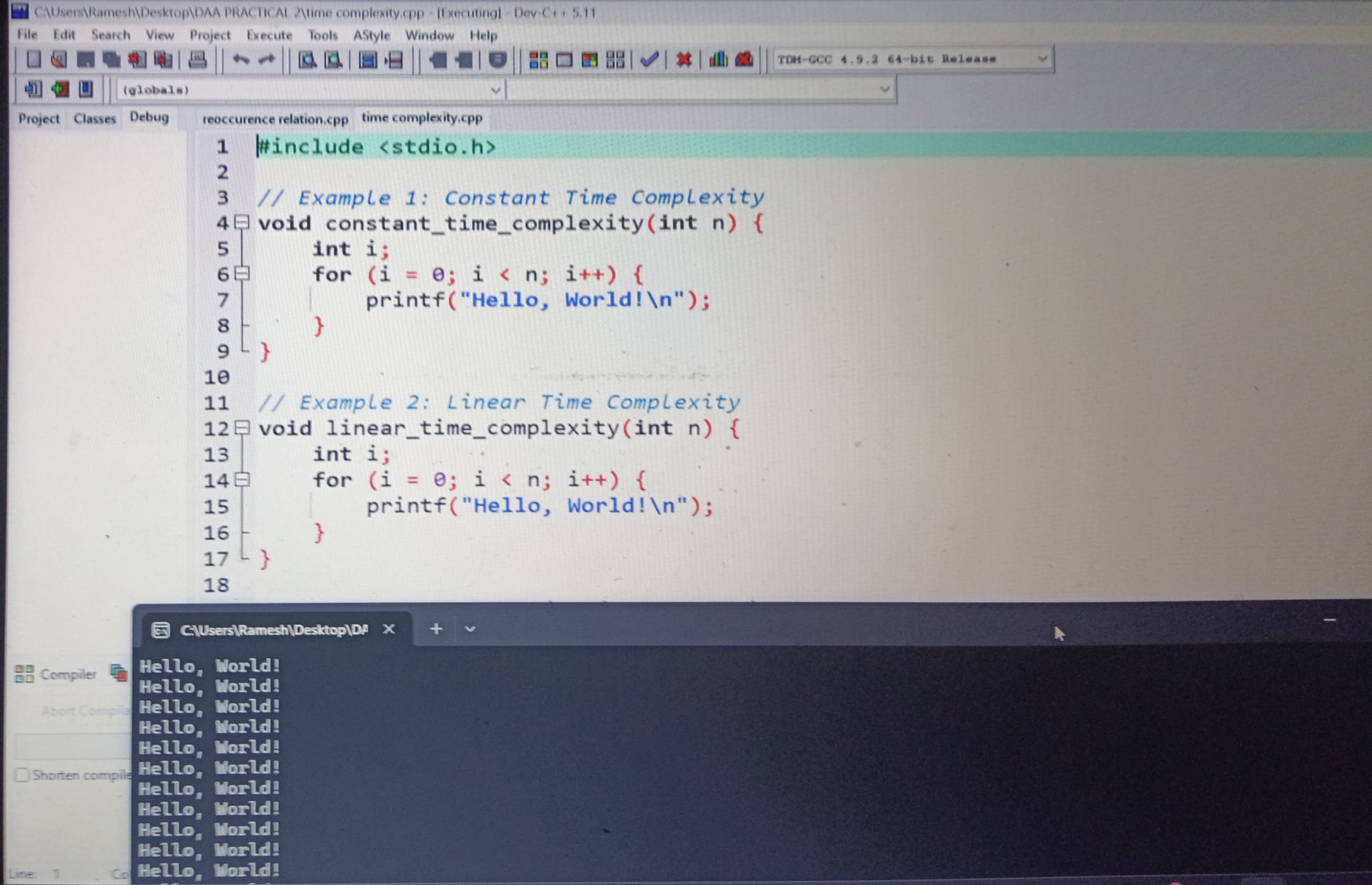
Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
asses Debug
reoccurence relation.cpp
time complexity.cpp

```c
#include <stdio.h>

// Example 1: Constant Time Complexity
void constant_time_complexity(int n) {
    int i;
    for (i = 0; i < n; i++) {
        printf("Hello, World!\n");
    }
}

// Example 2: Linear Time Complexity
void linear_time_complexity(int n) {
    int i;
    for (i = 0; i < n; i++) {
        printf("Hello, World!\n");
    }
}

// Example 3: Quadratic Time Complexity
void quadratic_time_complexity(int n) {
```

piler Resources Compile Log Debug Find Results Close

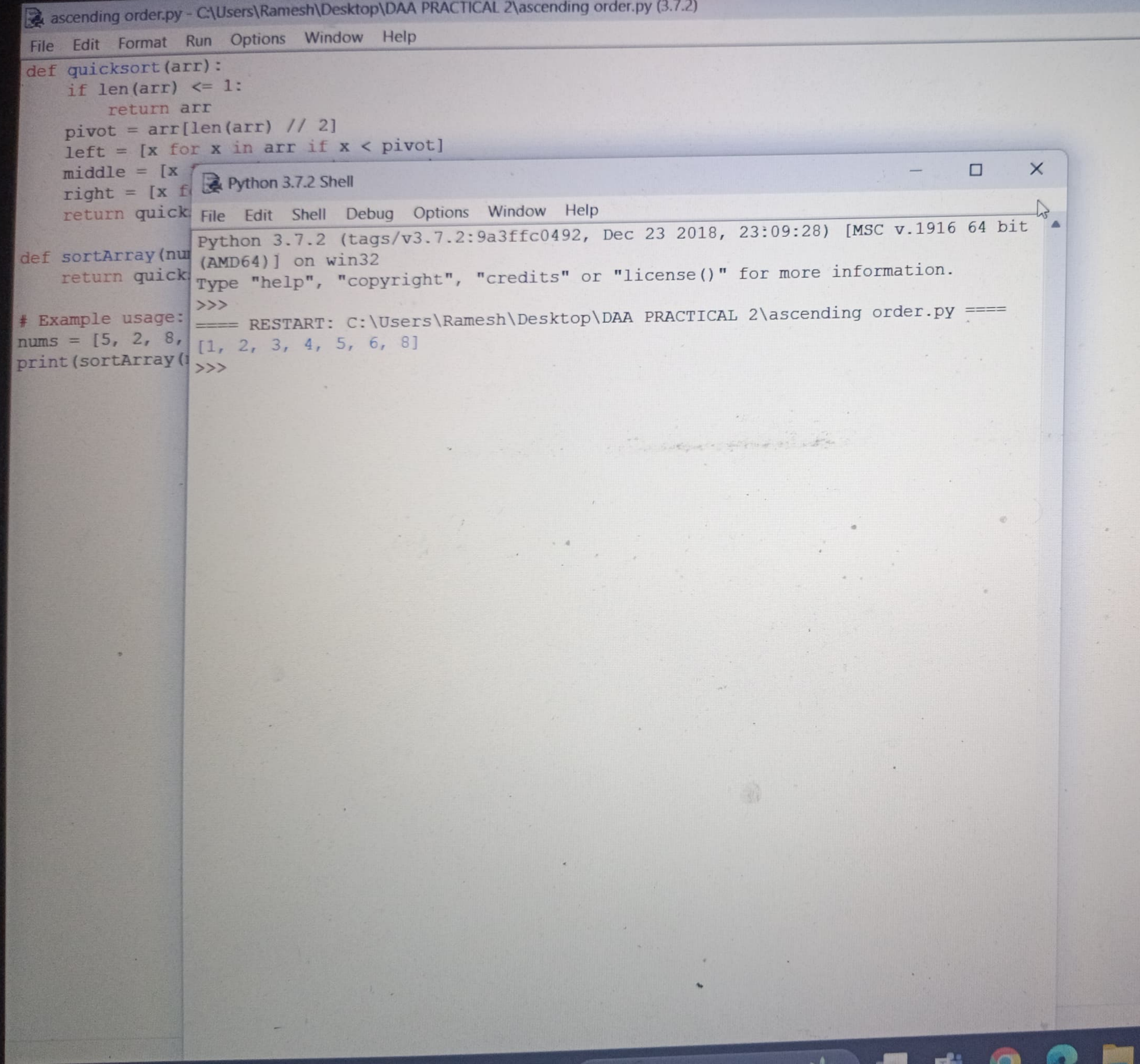
```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\reoccurence relation.exe
- Output Size: 129,251953125 KiB
- Compilation Time: 0,31s
```

```
Col: 1    Sel: 0    Lines: 60    Length: 1271    Insert    Done parsing in 0.016 seconds
```


C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\time complexity.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
reoccurence relation.cpp
time complexity.cpp
C:\Users\Ramesh\Desktop\DA
Compiler
Shorten compile

```
#include <stdio.h>

// Example 1: Constant Time Complexity
void constant_time_complexity(int n) {
    int i;
    for (i = 0; i < n; i++) {
        printf("Hello, World!\n");
    }
}

// Example 2: Linear Time Complexity
void linear_time_complexity(int n) {
    int i;
    for (i = 0; i < n; i++) {
        printf("Hello, World!\n");
    }
}

```

```
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
```

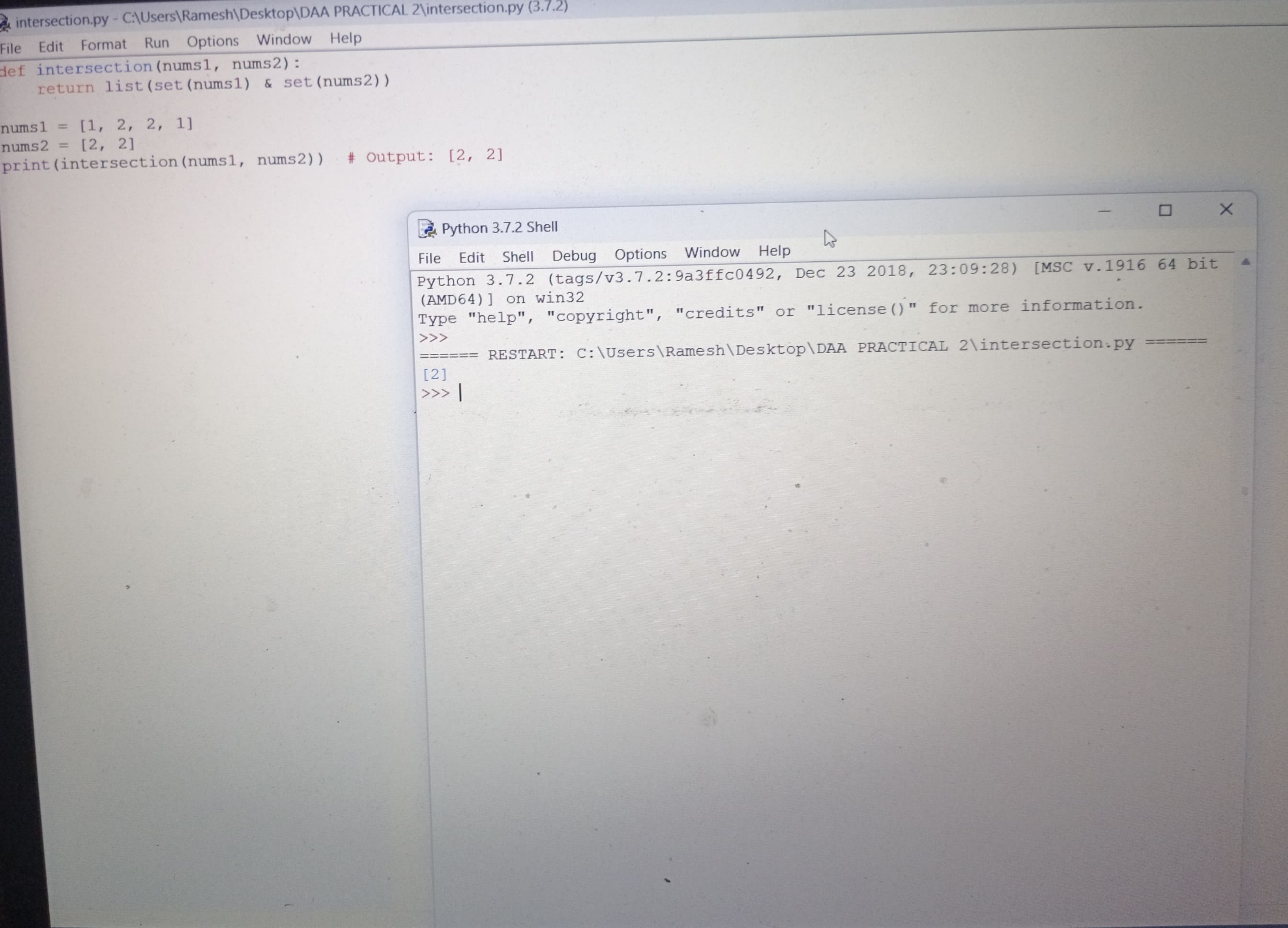


ascending order.py - C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\ascending order.py (3.7.2)

```
def quicksort(arr):
    if len(arr) <= 1:
        return arr
    pivot = arr[len(arr) // 2]
    left = [x for x in arr if x < pivot]
    middle = [x
    right = [x f
    return quick

def sortArray(nu
    return quick

# Example usage:
nums = [5, 2, 8,
print(sortArray(
```

Python 3.7.2 Shell

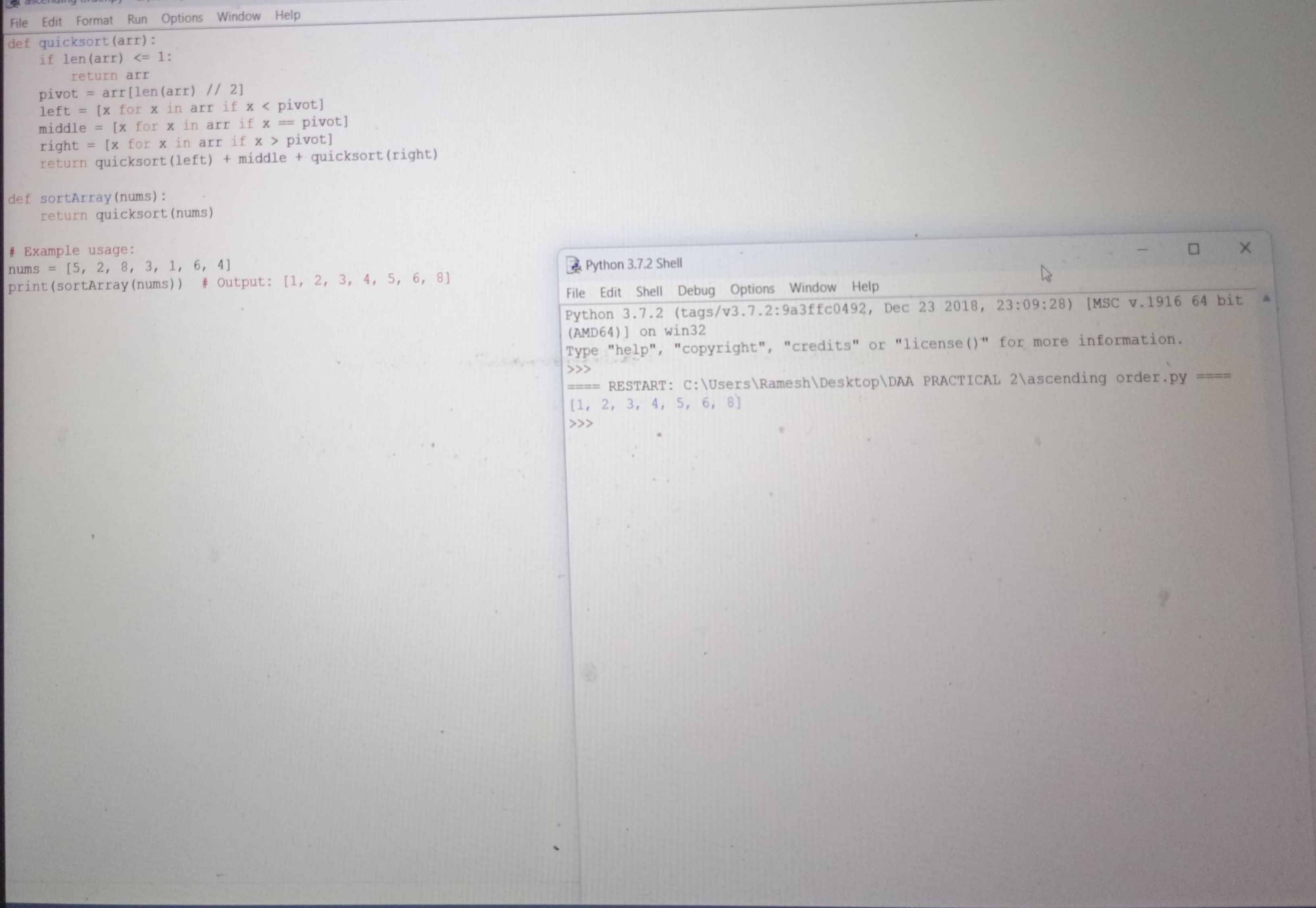
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\ascending order.py ====
[1, 2, 3, 4, 5, 6, 8]
>>>
```


intersection.py - C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\intersection.py (3.7.2)
File Edit Format Run Options Window Help

```
def intersection(nums1, nums2):
    return list(set(nums1) & set(nums2))

nums1 = [1, 2, 2, 1]
nums2 = [2, 2]
print(intersection(nums1, nums2))  # Output: [2, 2]
```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

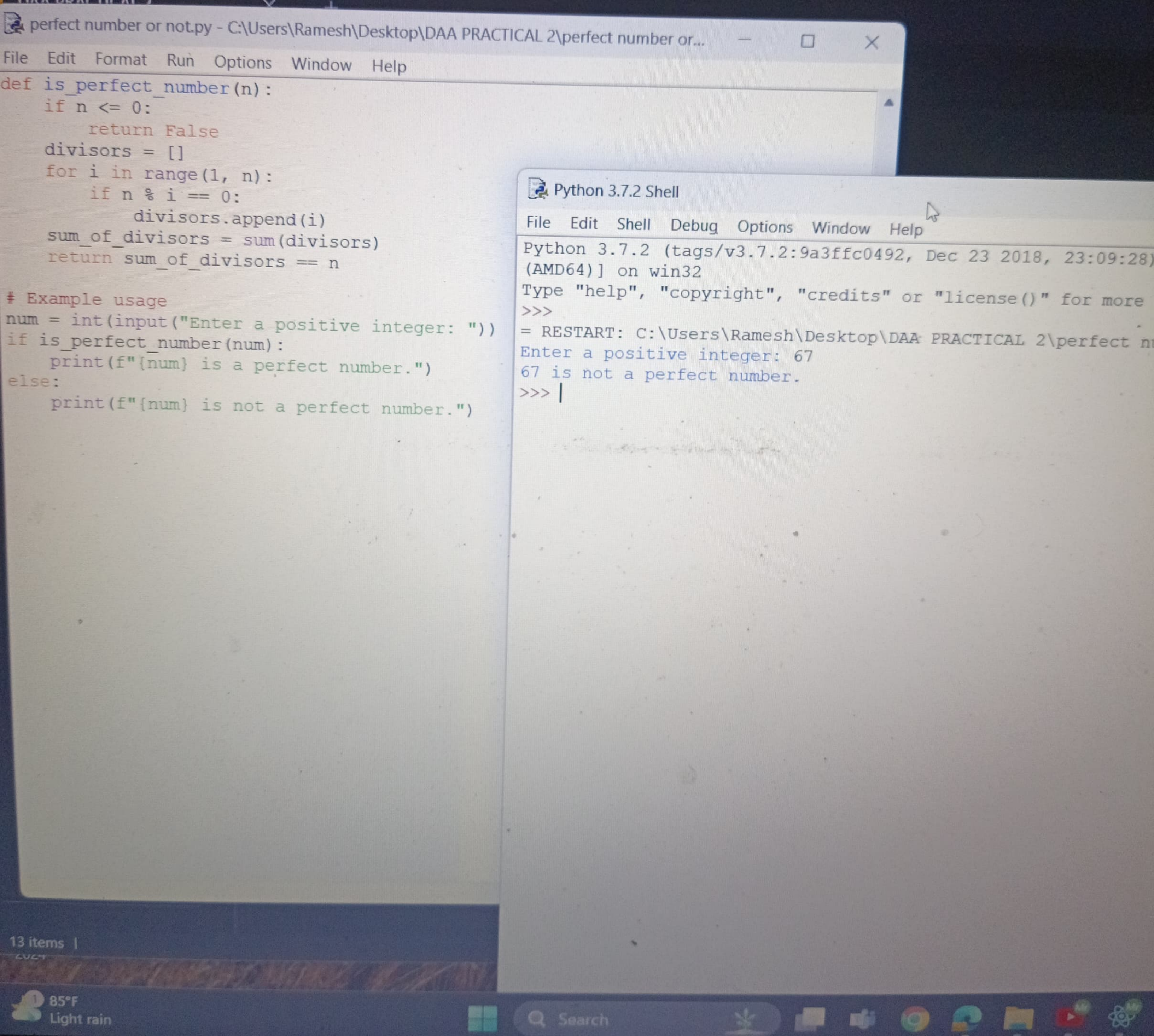
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
====== RESTART: C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\intersection.py ======
[2]
>>> 
```


File Edit Format Run Options Window Help

```
def quicksort(arr):
    if len(arr) <= 1:
        return arr
    pivot = arr[len(arr) // 2]
    left = [x for x in arr if x < pivot]
    middle = [x for x in arr if x == pivot]
    right = [x for x in arr if x > pivot]
    return quicksort(left) + middle + quicksort(right)

def sortArray(nums):
    return quicksort(nums)

# Example usage:
nums = [5, 2, 8, 3, 1, 6, 4]
print(sortArray(nums))  # Output: [1, 2, 3, 4, 5, 6, 8]
```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

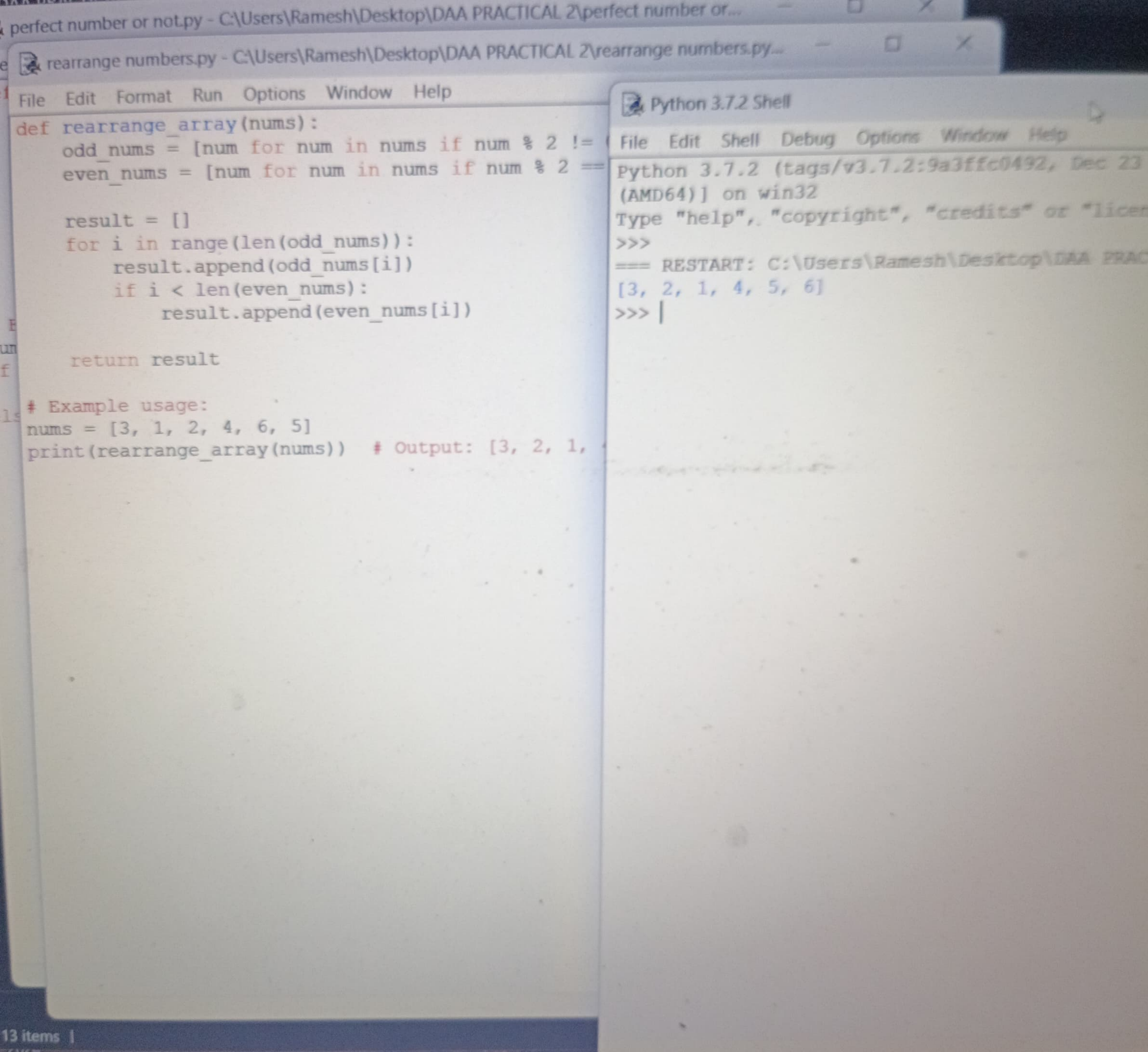
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\ascending order.py ====
[1, 2, 3, 4, 5, 6, 8]
>>>
```


perfect number or not.py - C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\perfect number or...
File Edit Format Run Options Window Help

```
def is_perfect_number(n):
    if n <= 0:
        return False
    divisors = []
    for i in range(1, n):
        if n % i == 0:
            divisors.append(i)
    sum_of_divisors = sum(divisors)
    return sum_of_divisors == n

# Example usage
num = int(input("Enter a positive integer: "))
if is_perfect_number(num):
    print(f"{num} is a perfect number.")
else:
    print(f"{num} is not a perfect number.")
```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28)
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more
>>>
= RESTART: C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\perfect nu
Enter a positive integer: 67
67 is not a perfect number.
>>>
```



perfect number or not.py - C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\perfect number or...

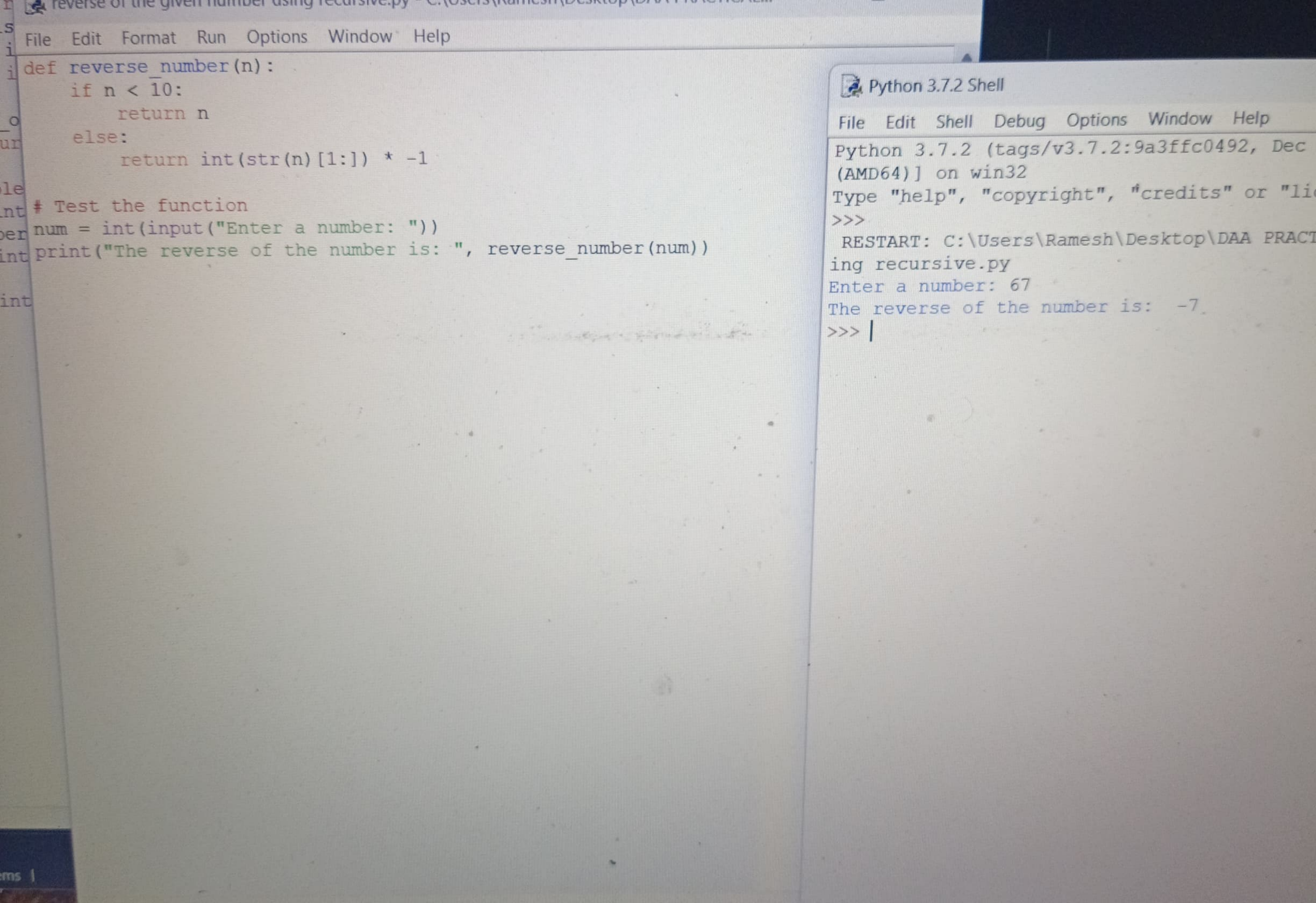
rearrange numbers.py - C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\rearrange numbers.py...

```
def rearrange_array(nums):
    odd_nums = [num for num in nums if num % 2 !=
    even_nums = [num for num in nums if num % 2 ==

    result = []
    for i in range(len(odd_nums)):
        result.append(odd_nums[i])
        if i < len(even_nums):
            result.append(even_nums[i])

    return result

# Example usage:
nums = [3, 1, 2, 4, 6, 5]
print(rearrange_array(nums))  # Output: [3, 2, 1,
```

Python 3.7.2 Shell

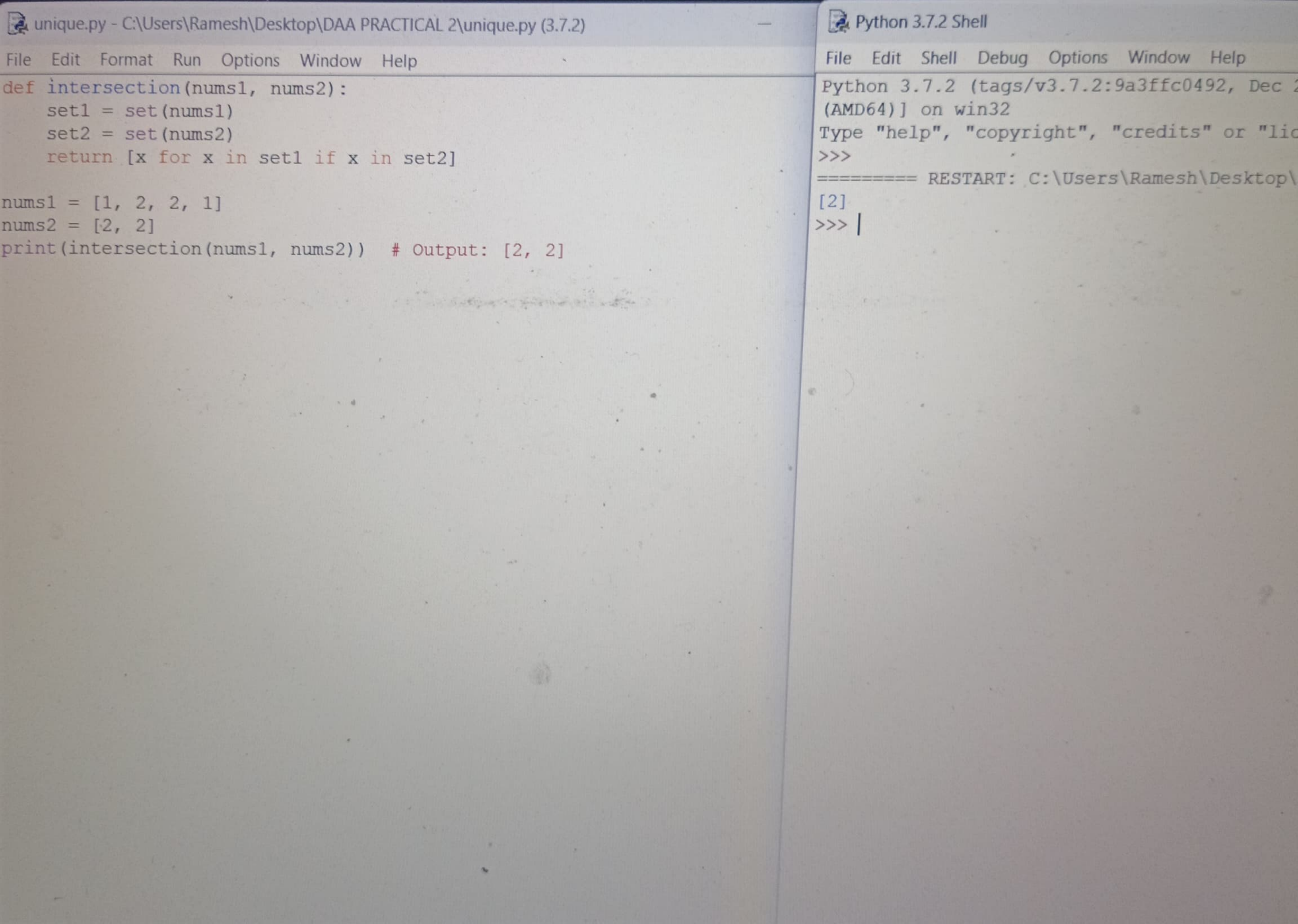
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23
(AMD64)] on win32
Type "help", "copyright", "credits" or "licen
>>>
=== RESTART: C:\Users\Ramesh\Desktop\DAA PRAC
[3, 2, 1, 4, 5, 6]
>>> 
```

File Edit Format Run Options Window Help
File Edit Shell Debug Options Window Help
13 items


File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
def reverse_number(n):
    if n < 10:
        return n
    else:
        return int(str(n)[1:]) * -1

# Test the function
num = int(input("Enter a number: "))
print("The reverse of the number is: ", reverse_number(num))
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec
(AMD64)] on win32
Type "help", "copyright", "credits" or "li
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA PRACT
ing recursive.py
Enter a number: 67
The reverse of the number is:  -7
>>> 
```


unique.py - C:\Users\Ramesh\Desktop\DAA PRACTICAL 2\unique.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```python
def intersection(nums1, nums2):
    set1 = set(nums1)
    set2 = set(nums2)
    return [x for x in set1 if x in set2]

nums1 = [1, 2, 2, 1]
nums2 = [2, 2]
print(intersection(nums1, nums2))  # Output: [2, 2]
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 
(AMD64)] on win32
Type "help", "copyright", "credits" or "lic
>>>
========= RESTART: C:\Users\Ramesh\Desktop\
[2]
>>> 
```