

NAME: ASIF ERFAN KHAN

ROLL NUMBER: 546

COURSE: MSc CS

SUBJECT: ALGORITHM

**TOPIC: SUBSET SUM
PROBLEM**

PRACTICAL 8

A recursive solution for subset sum

problem

Returns true if there is a subset

of set[] with sum equal to given sum

```
def isSubsetSum(set, n, sum):
```

```
    # Base Cases
```

```
    if (sum == 0):
```

```
        return True
```

```
    if (n == 0):
```

```
        return False
```

```
    # If last element is greater than
```

```
    # sum, then ignore it
```

```
    if (set[n - 1] > sum):
```

```
        return isSubsetSum(set, n - 1, sum)
```

```
    # else, check if sum can be obtained
```

```
    # by any of the following
```

```
    # (a) including the last element
```

```
    # (b) excluding the last element
```

```
    return isSubsetSum(
```

```
        set, n-1, sum) or isSubsetSum(
```

```
        set, n-1, sum-set[n-1])
```

```
# Driver code
```

```
set = [3, 34, 4, 12, 5, 2]

sum = 9

n = len(set)

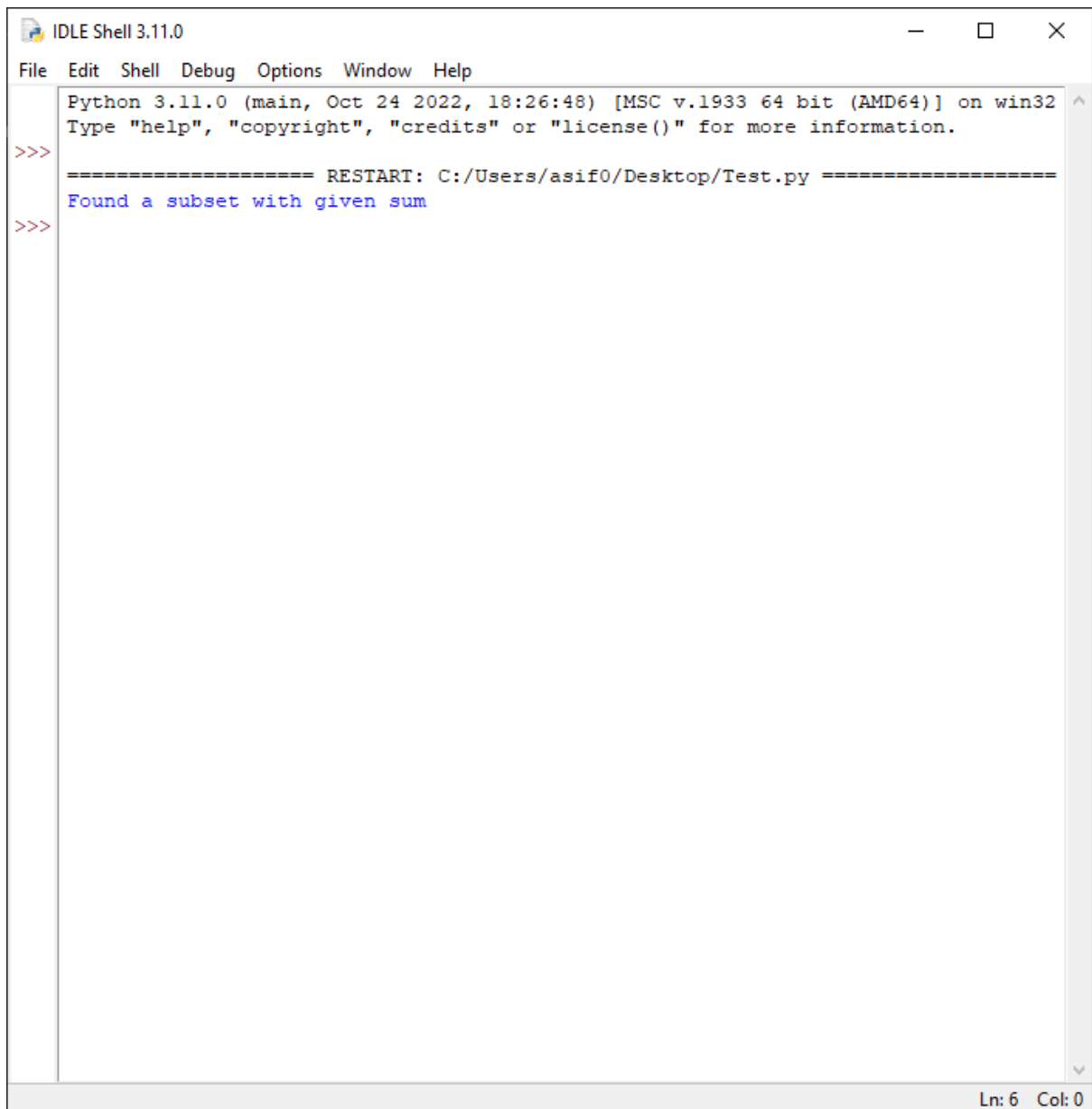
if (isSubsetSum(set, n, sum) == True):

    print("Found a subset with given sum")

else:

    print("No subset with given sum")
```

OUTPUT:



```
IDLE Shell 3.11.0
File Edit Shell Debug Options Window Help
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/asif0/Desktop/Test.py =====
Found a subset with given sum
>>>
```

Ln: 6 Col: 0