

Jacob Tucker

CMPS 390

3/9/23

Recursive Summation

The following screenshot shows my implementation and output for a recursive function that calculates the consecutive sum of a given number. It's base case is when the variable "n" is equal to 1. In that case, it simply returns 1. Otherwise it returns the sum of n and a recursive call to itself with 1 less than is its argument. For example, for the input 3 it will return:

consecutiveSum(3)

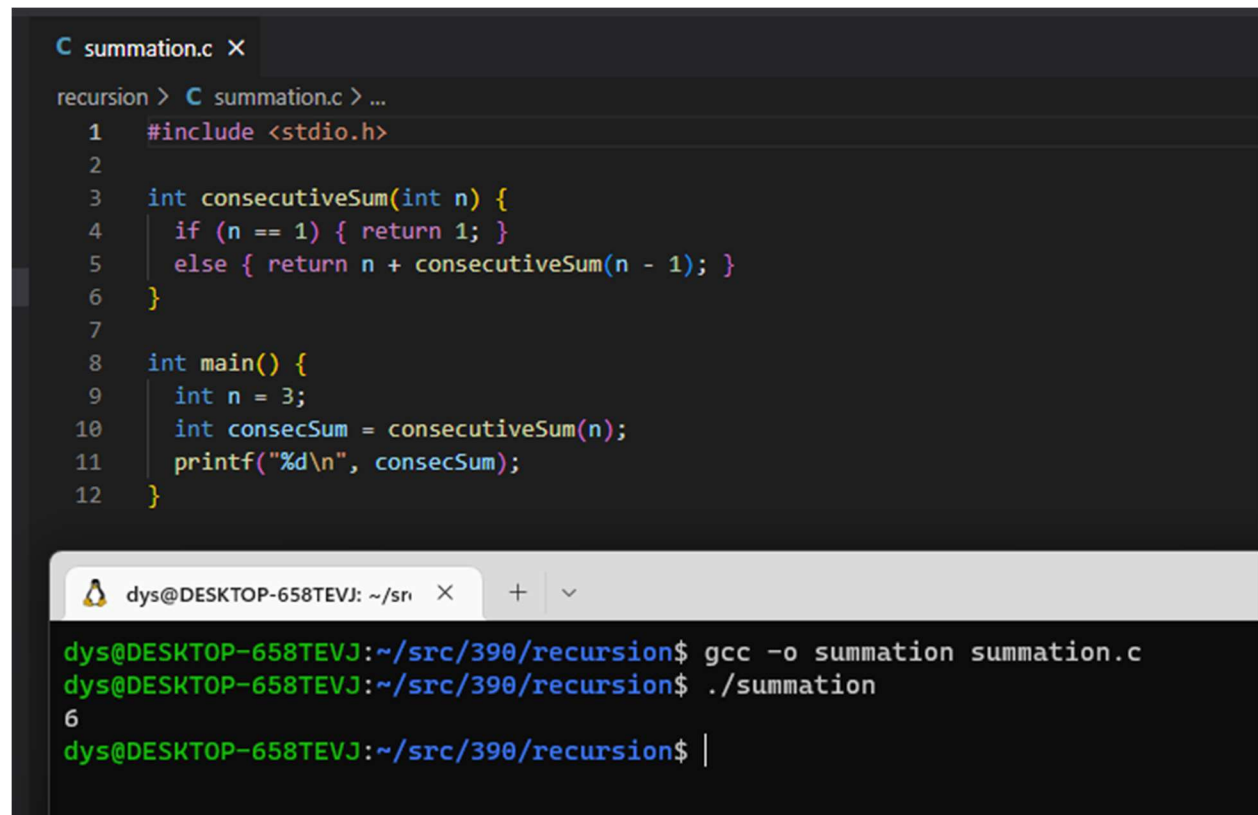
= 3 + consecutiveSum(2)

= 3 + 2 + consecutiveSum(1)

= 3 + 2 + 1

= 6

Therefore, this computes the consecutive sum correctly.



```
C summation.c X
recursion > C summation.c > ...
1  #include <stdio.h>
2
3  int consecutiveSum(int n) {
4      if (n == 1) { return 1; }
5      else { return n + consecutiveSum(n - 1); }
6  }
7
8  int main() {
9      int n = 3;
10     int consecSum = consecutiveSum(n);
11     printf("%d\n", consecSum);
12 }
```

```
dys@DESKTOP-658TEVJ: ~/src/390/recursion$ gcc -o summation summation.c
dys@DESKTOP-658TEVJ: ~/src/390/recursion$ ./summation
6
dys@DESKTOP-658TEVJ: ~/src/390/recursion$ |
```