

8/28/2023

Homework < 0

$$1) \sum_{i=1}^n i = \frac{n(n+1)}{2} \quad \frac{n+1(n+1+1)}{2}$$

LS: $\sum_{i=1}^1 1 = 1 \checkmark$

RS: $\sum_{i=1}^1 \frac{1(1+1)}{2} = \frac{2}{2} = 1 \checkmark$

$$\begin{aligned} \sum_{i=1}^n i &= \sum_{i=1}^{n-1} i + n \\ &= \frac{n(n-1)}{2} + (n-1) \frac{2}{2} \\ &= \frac{n(n-1)}{2} + \frac{2(n-1)}{2} \\ &= \frac{n(n-1) + 2n-2}{2} \\ &= \frac{n^2 - n + 2n - 2}{2} \\ &= \frac{n^2 + n - 2}{2} \\ &= \frac{(n+1)(n-2)}{2} \\ &= \boxed{\frac{n+1(n+1-1)}{2}} \end{aligned}$$

2)

a) Base cases: you must always have a base case, that can be solved w/o recursion

b) Making progress: for cases solved recursively, the call must always be a case that progresses to a base case

c) Design Rule: assume all recursive calls work

d) Compound Interest rule: never duplicate work by solving the same instance of the problem in separate recursive calls

Homework 0

3) No, the function will only terminate for all inputs that are positive and zero. If a negative number is inputted the function will run to 0 but not terminate since the instance of $i \neq 0$ never occurs.

4) class Person {

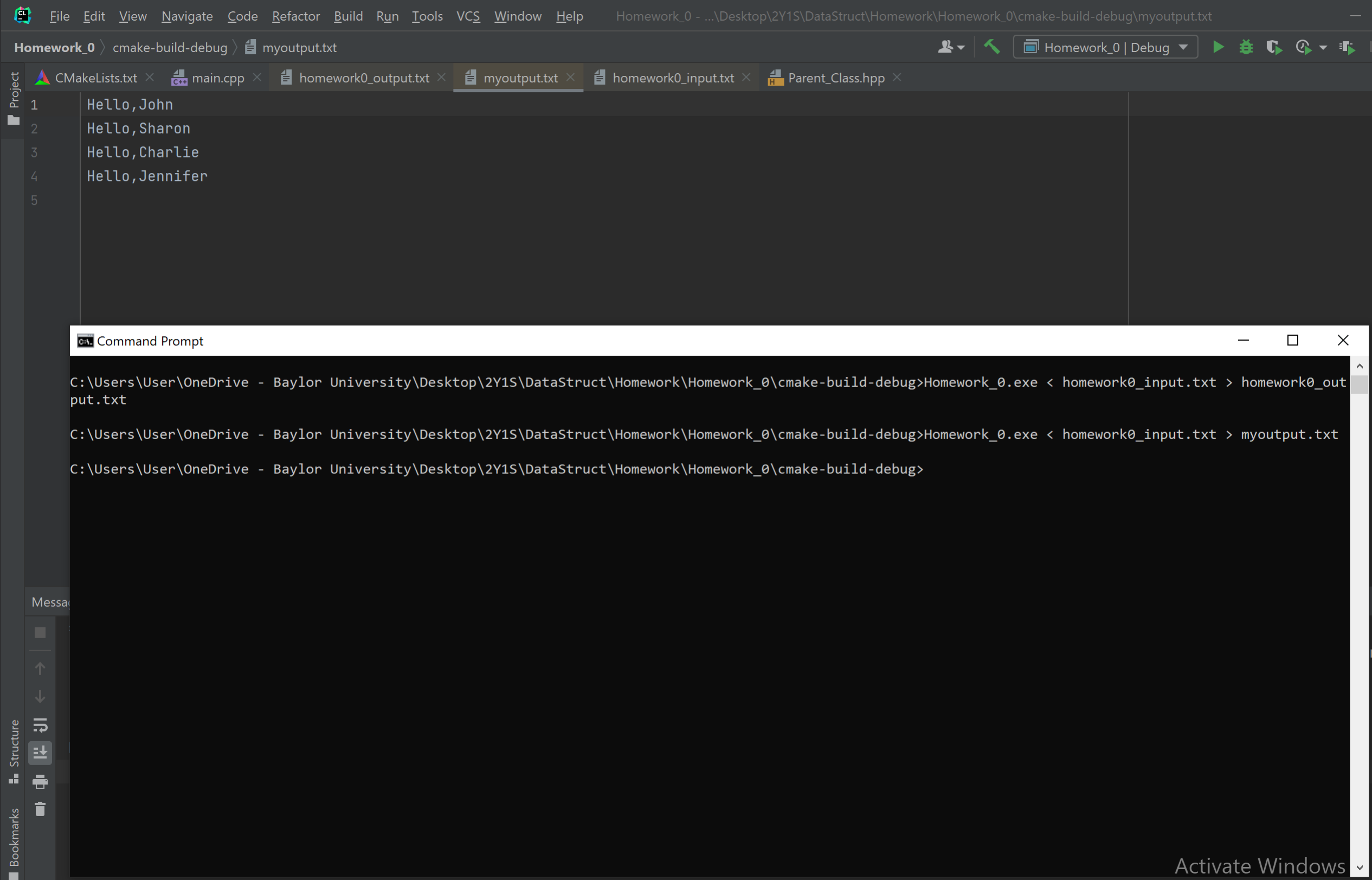
protected:

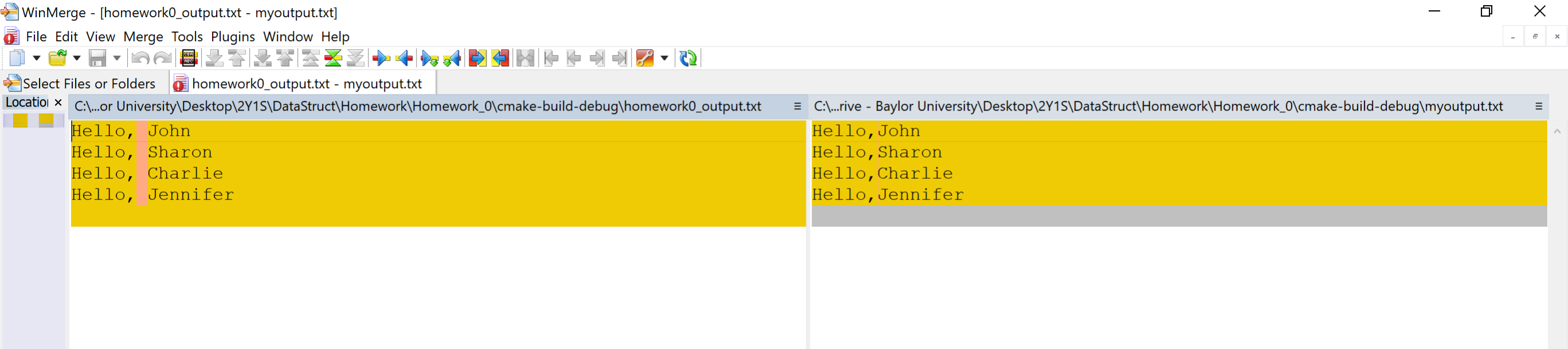
string name;

string phone number;

public:

```
10 using namespace std;
11
12 class Person {
13     protected:
14         string name;
15         string phoneNumber;
16 };
17
18 class Employee:: public Person {
19     private:
20         int employeeNumber;
21 }
22 class Student:: public Person {
23     private:
24         int studentID;
25 };
26
27
28 #endif //HOMEWORK_0_PARENT_CLASS_HPP
29
```






```
1 Hello, John
2 Hello, Sharon
3 Hello, Charlie
4 Hello, Jennifer
5
```

Command Prompt

```
C:\Users\User\OneDrive - Baylor University\Desktop\2Y1S\DataStruct\Homework\Homework_0\cmake-build-debug>Homework_0.exe < homework0_input.txt > myoutput.txt
```

```
C:\Users\User\OneDrive - Baylor University\Desktop\2Y1S\DataStruct\Homework\Homework_0\cmake-build-debug>
```

