

# Midterm Part 2

CS-102

Fall 2021

# Program Specifications

- **Carefully read the entire Program Specification before beginning.**
- FOLLOW THE SPECIFICATIONS BELOW TO create the following functions to calculate and display the sum or product of the integers from 1 to the number input.
- **getNum**
  - Receives no input. Prompts the user to enter an integer. Uses a loop to validate. Insures that no number, entered, is negative or so large that the resulting product becomes negative. Returns the result.
- **calcSum**
  - Receives an integer as input. USE AN ACCUMULATOR to calculate the sum of the integers from 1 to this number. Returns the result.
- **calcProd**
  - Receives an integer as input. USE AN ACCUMULATOR to calculate the product of the integers from 1 to this number. Returns the result.

# Program Specifications

- Main should:
  - Prompt for a character that determines if you want to calculate the sum ('S' or 's') or product ('P' or 'p'). Use a switch to process the choices. Upper and lower case values should work the same way. Use default to give an error message if anything else is entered.
  - If a valid option is entered, use getNum to prompt for the number
  - **Note that calcSum and calcProduct RETURN the result to main, and the output is printed in main.**
- For example, if you input 5 as the number, and 'S', the program should calculate  $1 + 2 + 3 + 4 + 5$  and display the sum as 15.
- If you input 5 and 'P' the program should calculate  $1 \times 2 \times 3 \times 4 \times 5$  and display the product as 120.
- The process should repeat as long as the user wants to continue. This loop should be in main.

# Sample Test Run

Enter S for sum, P for prod: **z**

Invalid choice

Again(y/n)? **y**

Enter S for sum, P for prod: **s**

Enter an integer greater than 1: **1**

Must be greater than 1, re-enter: **-3**

Must be greater than 1, re-enter: **3**

The sum of the numbers from 1 to 3 is 6

Again(y/n)? **y**

Enter S for sum, P for prod: **p**

Enter an integer greater than 1: **4**

The product of the numbers from 1 to 4 is 24

Again(y/n)? **n**

# Strategy

- There are four functions in this program:
  1. main
  2. getNum
  3. calcSum
  4. calcProd
- Remember, you are not awarded any points for programs that do not compile or run. So, your best strategy is to make each of the three functions work before including them in the program.
- Only one program should be submitted, so you want to be sure that the program you submit works, even if it is not complete.
- For example, you might start by getting main() to call getNum() and have it return a number that you enter.
- Then you might add the ability to do calcSum().

# Strategy

- Then you may add the capability to do calcProd() to your program.
- The important thing is that whatever you turn in, even if not complete in functionality, should run and show what you have been able to accomplish.
- Points will be awarded as follows:
  - 3 points - Making program run getNum() correctly.
  - 3 points – Making calcSum() run (assuming getNum() is already running).
  - 3 points – Making calcProd() run (assuming getNum() is already running).
  - 2 points – Making main() run with all the functionality required in the program specifications.
  - 1 point – Formatting the program to meet the Gaddis standards we have been using.
- Call your program ***MyName-Midterm.cpp*** .
- **Program submission must be made no later than 6:00pm.** Programs submitted after 5:59pm will not be accepted by Canvas.