****

**Department of Computer Science**

**HITEC University, Taxila**

**BS Computer Science Program**

**(Batch 2022)**

**CS-206 Computer Organization & Assembly language 4(3+1)**

**Lab Folder**

**SPRING 2024**

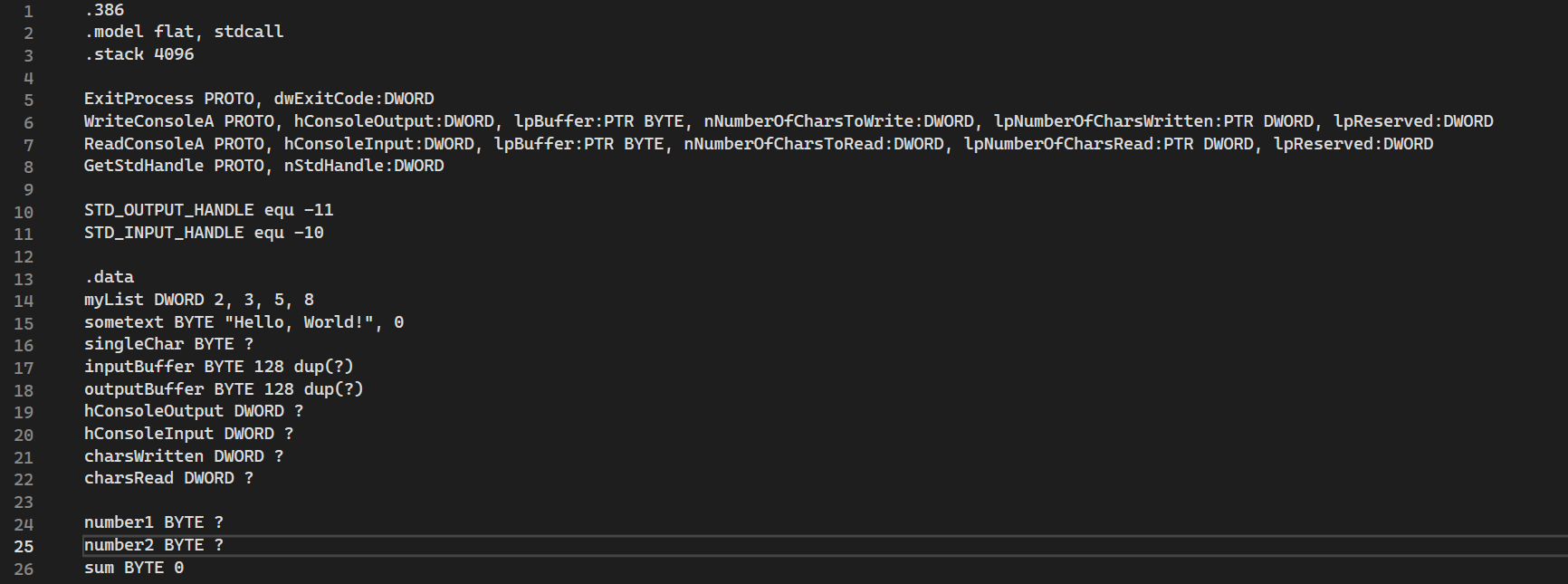
**Instructor: Fatima Rauf**

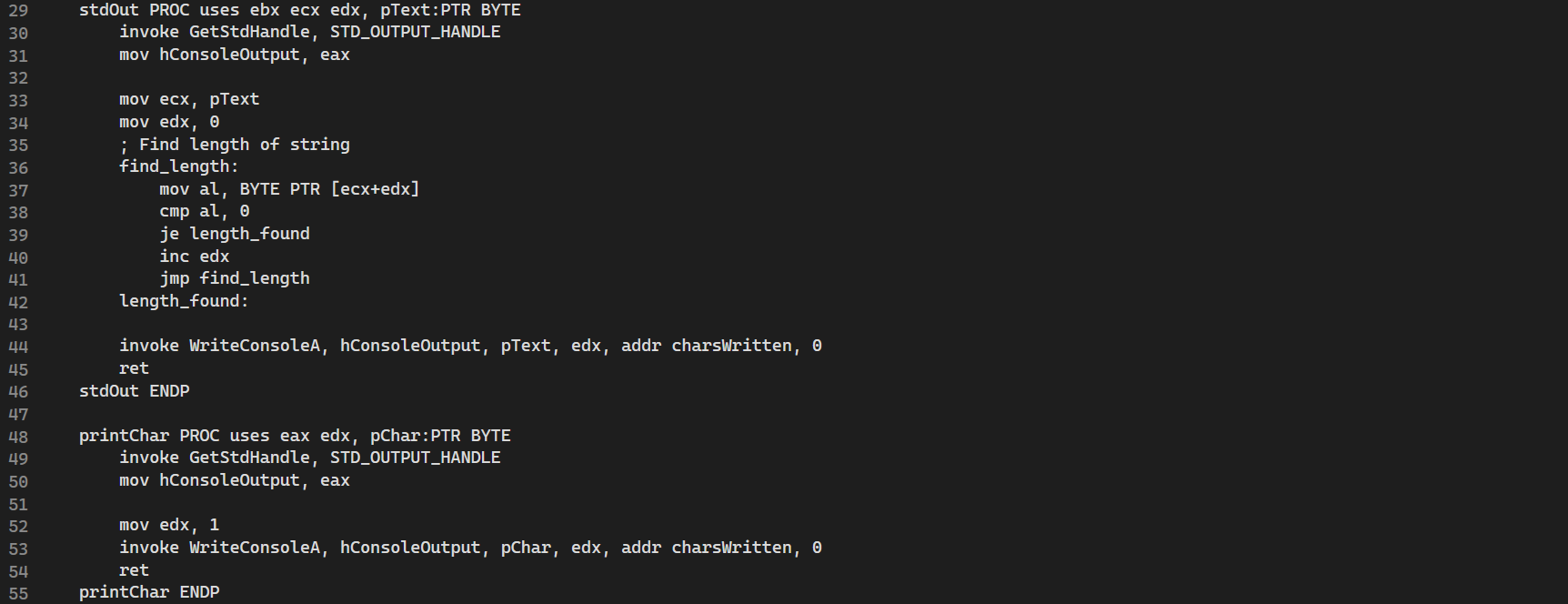
**Submitted By:**

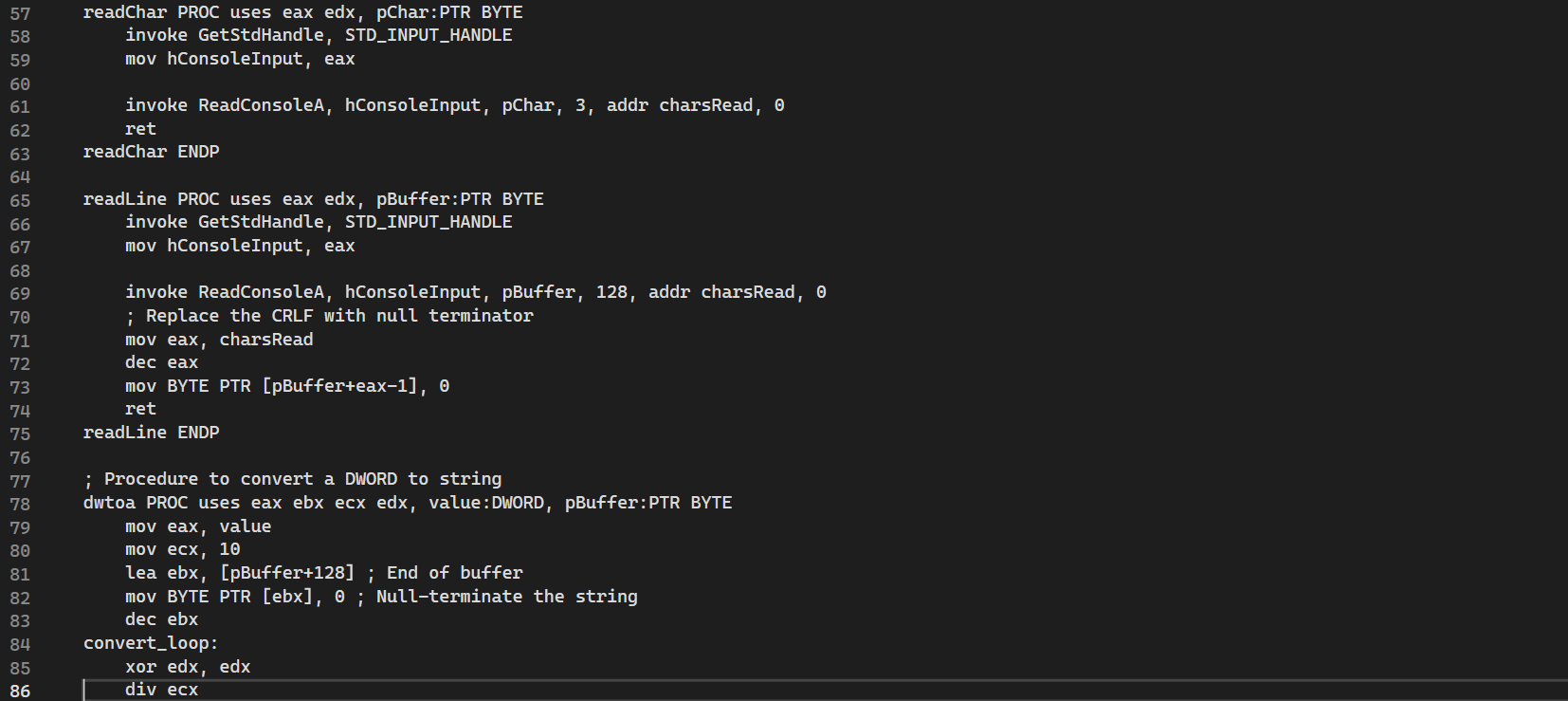
**Abdul Ahad (22-CS-071)**

**Lab Report # 12**

**Pre-Definition’s:**

****

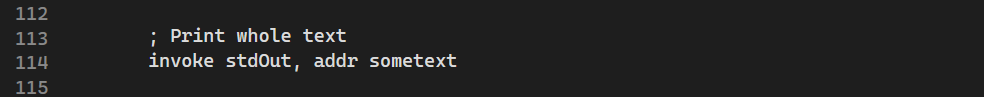
****

****

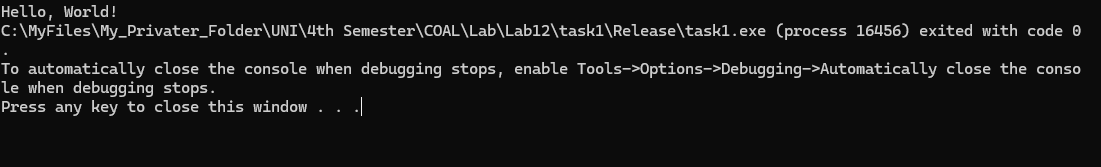
****

**Task 1:** Write code in MASAM that displays a message.

**Code:**

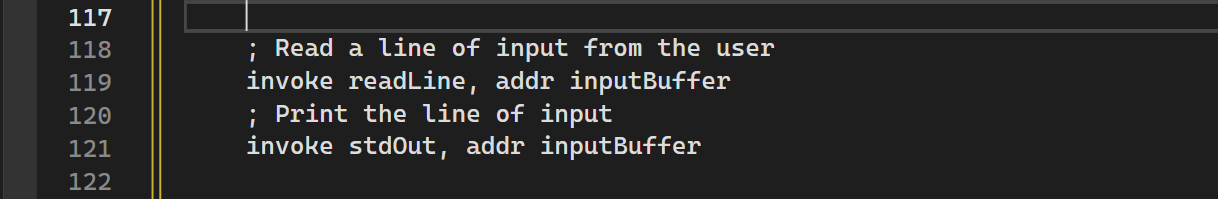
****

**Output:**

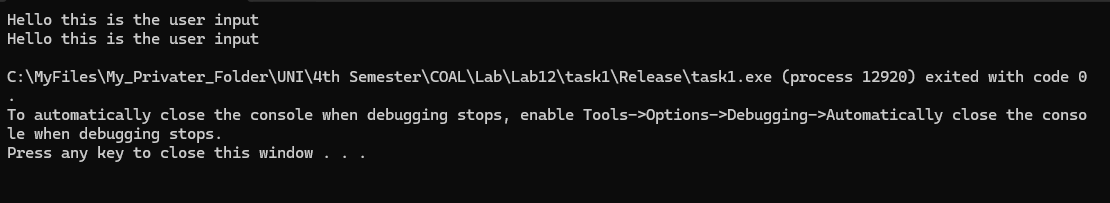
****

**Task 2:** Write code in MASAM that inputs a message and display it.

**Code:**

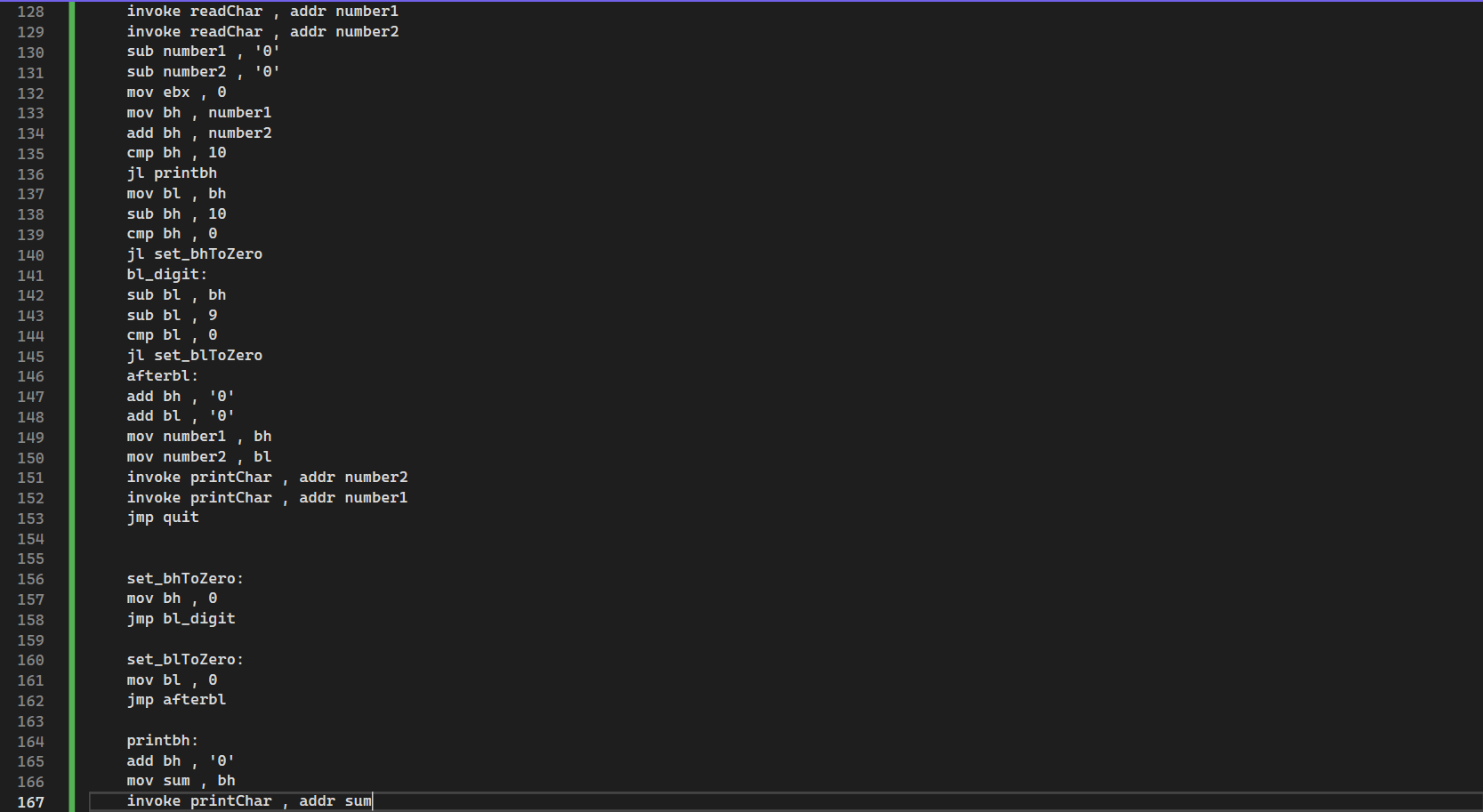
****

**Output:**

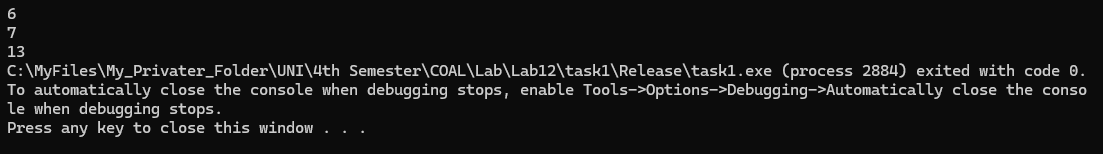
****

**Task 3:** Write code in MASAM for addition of two user defined numbers.

**Code:**

****

**Output:**

****

**Conclusion:**

This assembly program demonstrates basic console input and output operations using x86 assembly language and Windows API functions. It includes several procedures for interacting with the console and performs a task involving reading characters, converting them to numbers, adding them, and printing the result. Here's a summary of its functionality:

1. **Setup and Declarations**:
   * The program sets up the stack and declares prototypes for Windows API functions used for console input/output (I/O).
   * It defines data sections with variables for console handles, buffers, and character counts, as well as a list of integers and a text string.
2. **Procedures for Console I/O**:
   * stdOut: Prints a null-terminated string to the console.
   * printChar: Prints a single character to the console.
   * readChar: Reads a single character from the console.
   * readLine: Reads a line of input from the console and null-terminates it.
   * dwtoa: Converts a DWORD value to a string representation.
   * printReg and printVal: Print the contents of a register or value after conversion to a string.
3. **Main Procedure**:
   * **Task 1**: Prints a predefined text string ("Hello, World!") to the console using the stdOut procedure.
   * **Task 2**: Reads a line of input from the user using the readLine procedure and prints it back to the console using the stdOut procedure.
   * **Task 3**:
     + Reads two characters from the console using the readChar procedure.
     + Converts these characters from ASCII to numerical values.
     + Adds the two numerical values.
     + If the sum is a single digit, it prints the result directly.
     + If the sum is a two-digit number, it handles the tens and units digits separately and prints both.
4. **Exit**:
   * The program ends by invoking the ExitProcess function to terminate the process.

The program showcases essential operations like console input and output, string manipulation, and simple arithmetic in assembly language. It highlights how to use low-level procedures to perform tasks that are typically straightforward in higher-level languages.