Salwaa Mumtaazah D.

23/516172/PA/22060

CSB

**Assignment 1**

1.6.2 First Task: C++ Code to Assembly

1. Write a Simple C++ Program

Write a C++ program that adds two integers.

A screenshot of a computer program

Description automatically generated

1.6.3 Second Task: Assembly to C++

1. Analyze the Provided Assembly Code

Consider the following assembly code (for illustration purposes; it may not compile directly):

section .data

num1 dw 5

num2 dw 10

result dw 0

section .text

global \_start

\_start:

mov ax, [num1]

imul ax, [num2]

mov [result], ax

; Exit the program

mov eax, 1

xor ebx, ebx

int 0x80

Explanation:

The code is divided into four sections: Data Segment, Code Segment, Execution Code, and Program Exit.

* **Data Segment**

This section defines the program's variables. The code defines two labels, num1 and num2, which store the values 5 and 10, respectively. The label result is initialized to 0 and stores the result of the multiplication.

* **Code Segment**

This section contains the executable code. The label \_start is defined as the entry point of the program, where execution begins.

* **Execution Code**

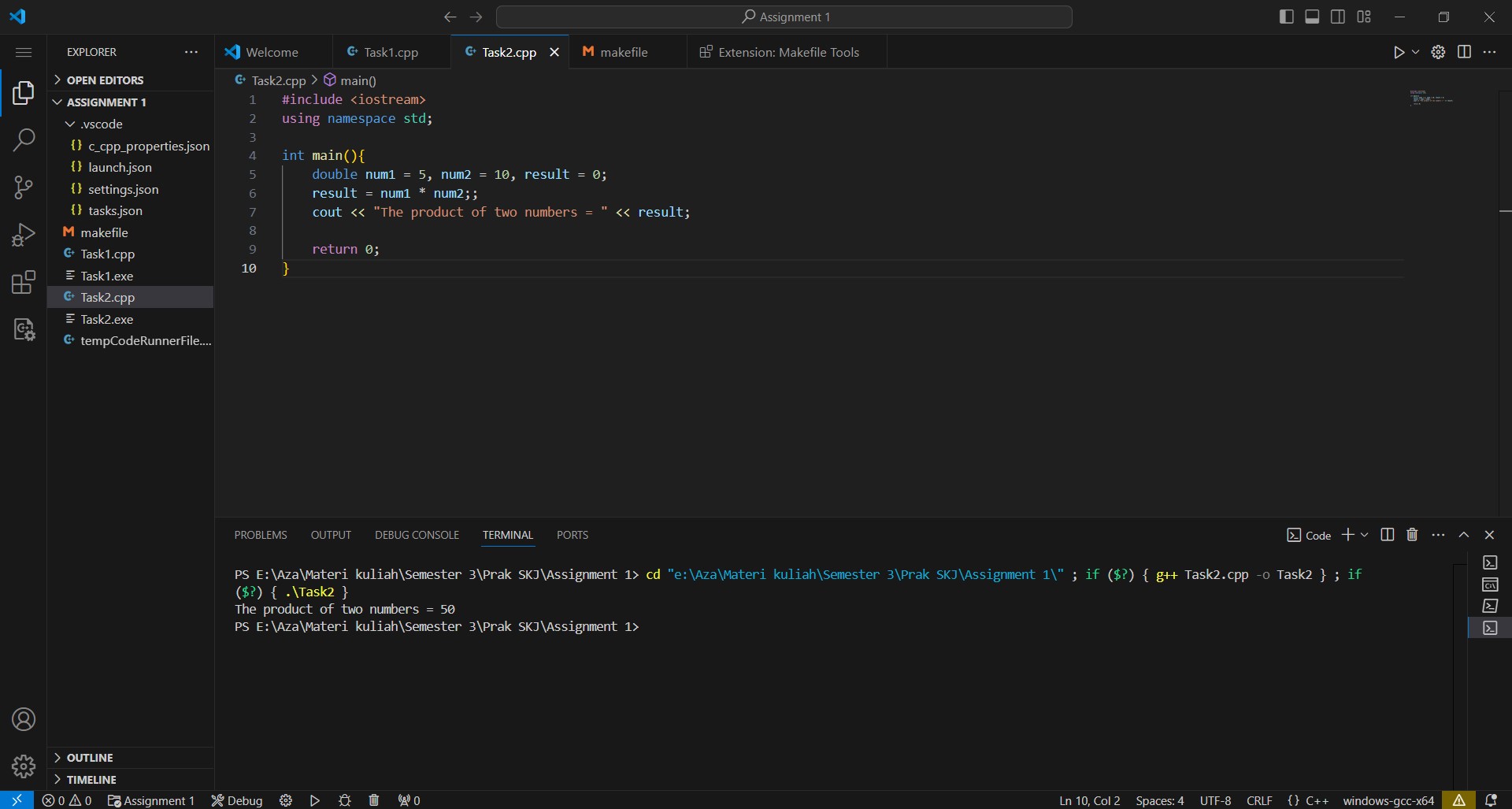
This section contains the instructions that perform the multiplication and store the result. The code loads the values of num1 and num2 into registers, multiplies them using the imul instruction, and stores the result in the result label. The result is then stored in memory.

* **Program Exit**

This section contains the instructions that exit the program. The code sets the exit status code to 0 and triggers a system call using the interrupt instruction int 0x80.

2. Write the Equivalent C++ Code (10 points)

Based on the provided assembly code, write a C++ program that performs the same functionality. The C++ program should produce the same result as the assembly code.



c