**Part 1: Writing the C Program**

1. **Task Description:** Write a C program that has an integer value and then performs a checking operation on this number whether it is positive or negative The program should print the result of the operation to the standard output.
2. **Sample Code:**

#include <stdio.h>

int main() {

int a = 55;

if(a>=0){printf("The number: %d is positive\n", a);}

else {printf("The number: %d is negative\n", a);}

return 0; }

**Part 2: Converting to Assembly**

1. **Task Description:** Use the **gcc** compiler to convert the C program into its assembly language representation.
2. **Instructions:**
   * Compile the C program to assembly using **gcc -S -o program.s program.c**, replacing **program.c** with the name of the C source file.
   * Explain the command: This **gcc** command compiles the C program into an assembly language file without assembling or linking. The **-S** flag tells **gcc** to stop after the compilation phase, producing an assembly file. The **-o** option specifies the output file name.

**Part 3: Modifying the Assembly Code**

1. **Task Description:** Modify the generated assembly code to change the arithmetic operation from addition to subtraction.

**Part 4: Compiling and Running the Modified Assembly Code**

1. **Task Description:** Compile the modified assembly code back into an executable program and run it to verify that it works as expected.
2. **Instructions:**
   * Compile the assembly code using **gcc -o modified\_program program.s**, replacing **program.s** with the modified assembly file name.
   * Run the compiled program to test the new functionality.