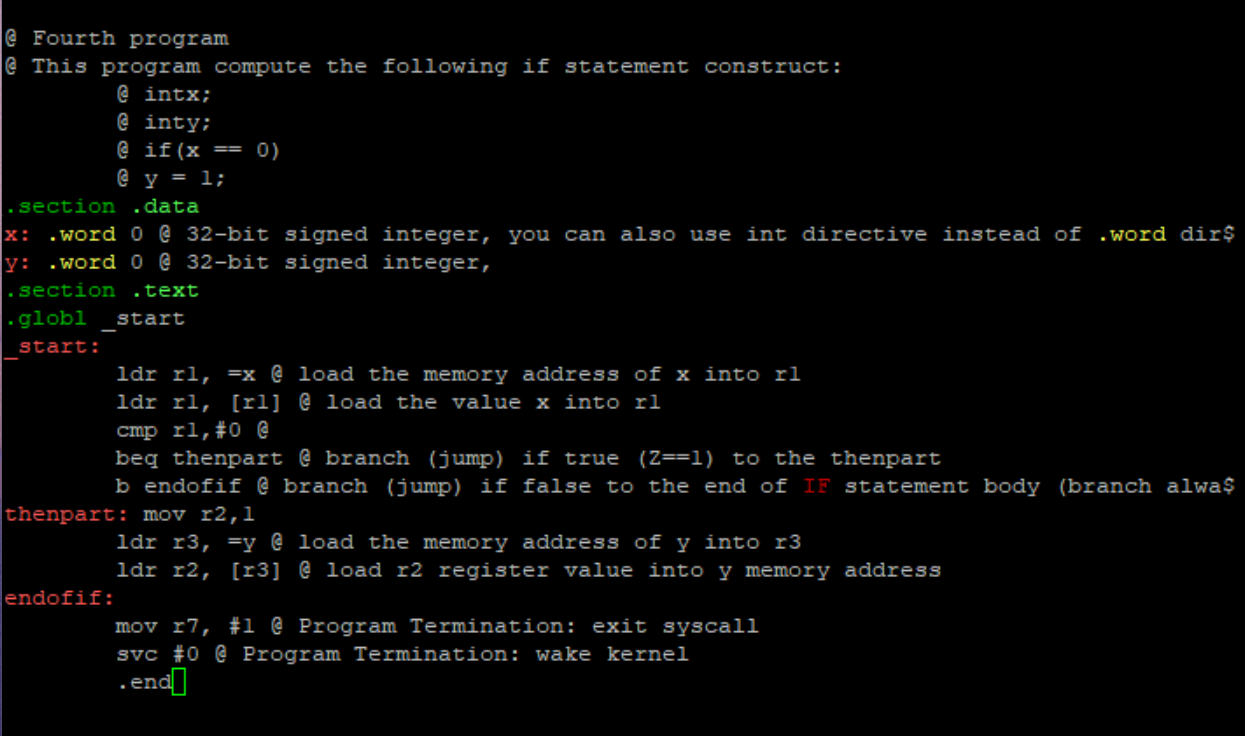
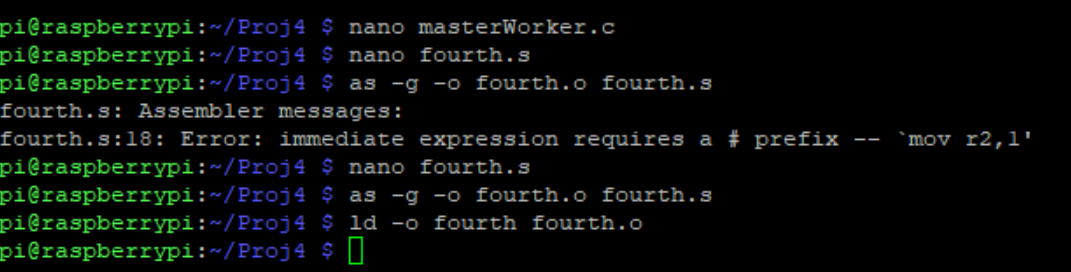
Task 4 Arm Assembly

Praveen Doluweera

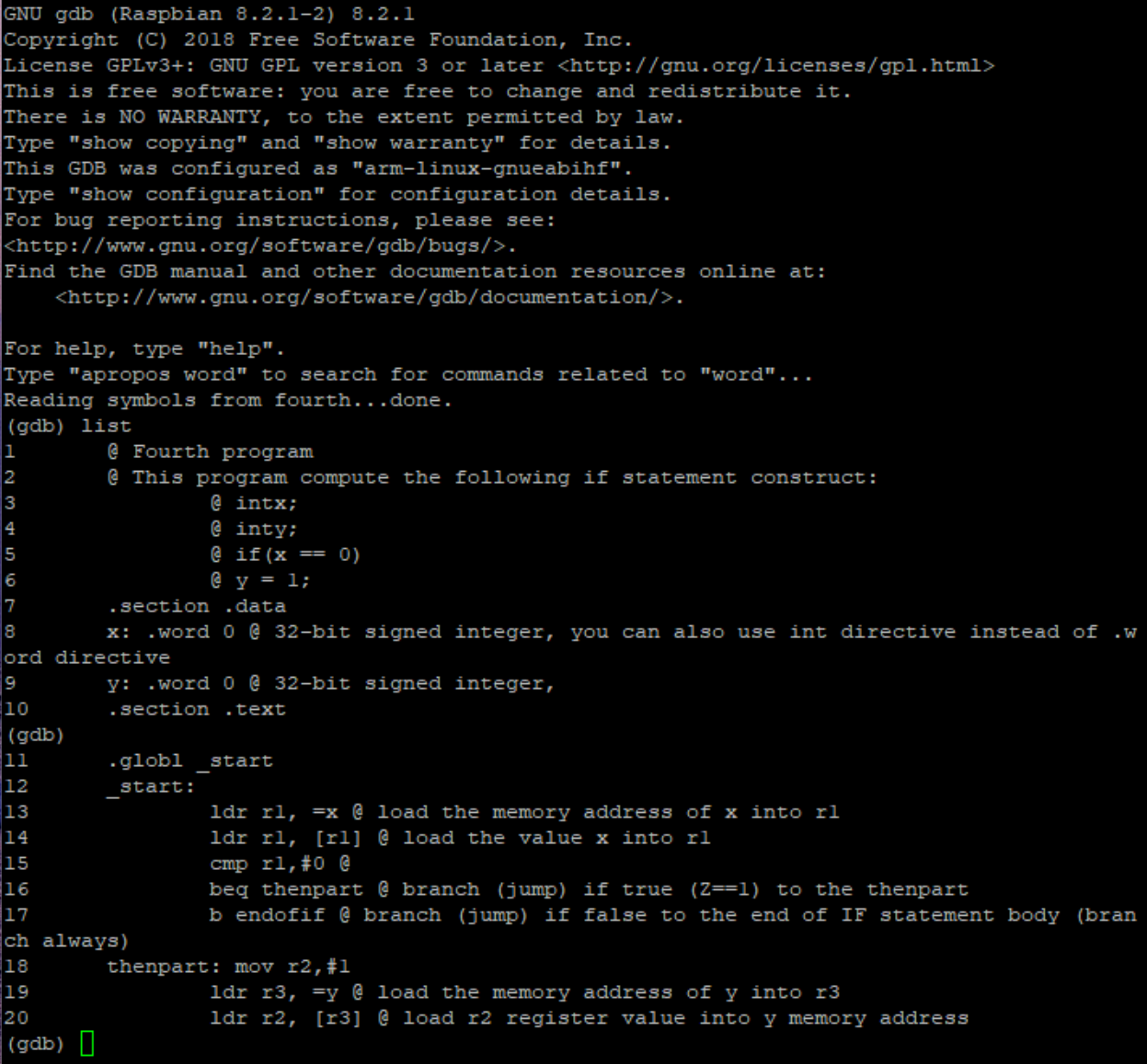


*Figure 1*

I first type out the program shown in Figure 1 using the nano command. There was an error as the # was not added in front of the immediate next to mov r2, 1. I fixed the problem and compiled the program as shown below.



*Figure 2*



*Figure 3*

The code is listed above in the debugger

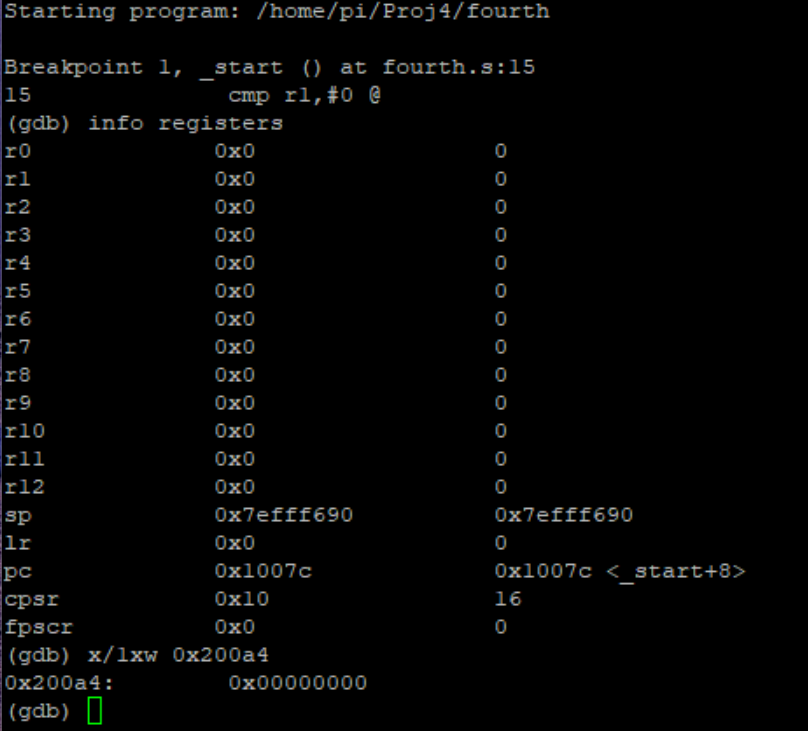


Figure 4

I check the memory address of x as shown above in figure 4

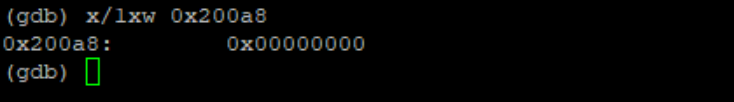
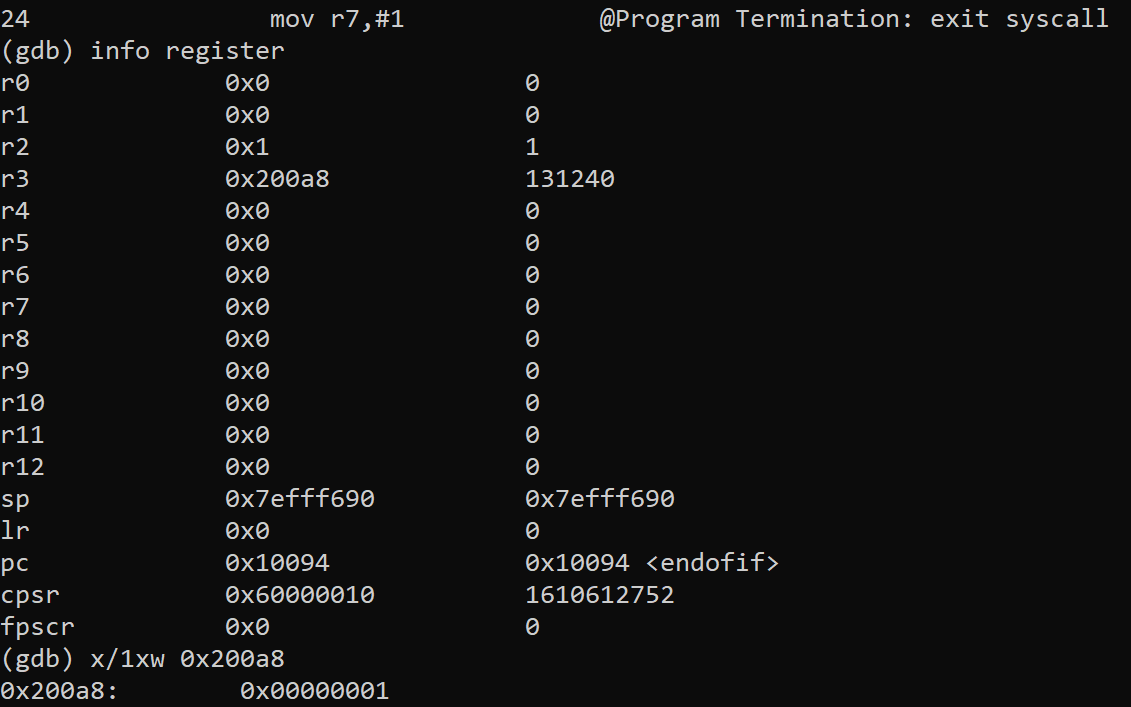


Figure 5

The memory address of y is shown above

*Figure 6*

Debugging the fourth program is shown above. The value is different because it jumps to the procedure called thenpart instead of executing the b endofif line.

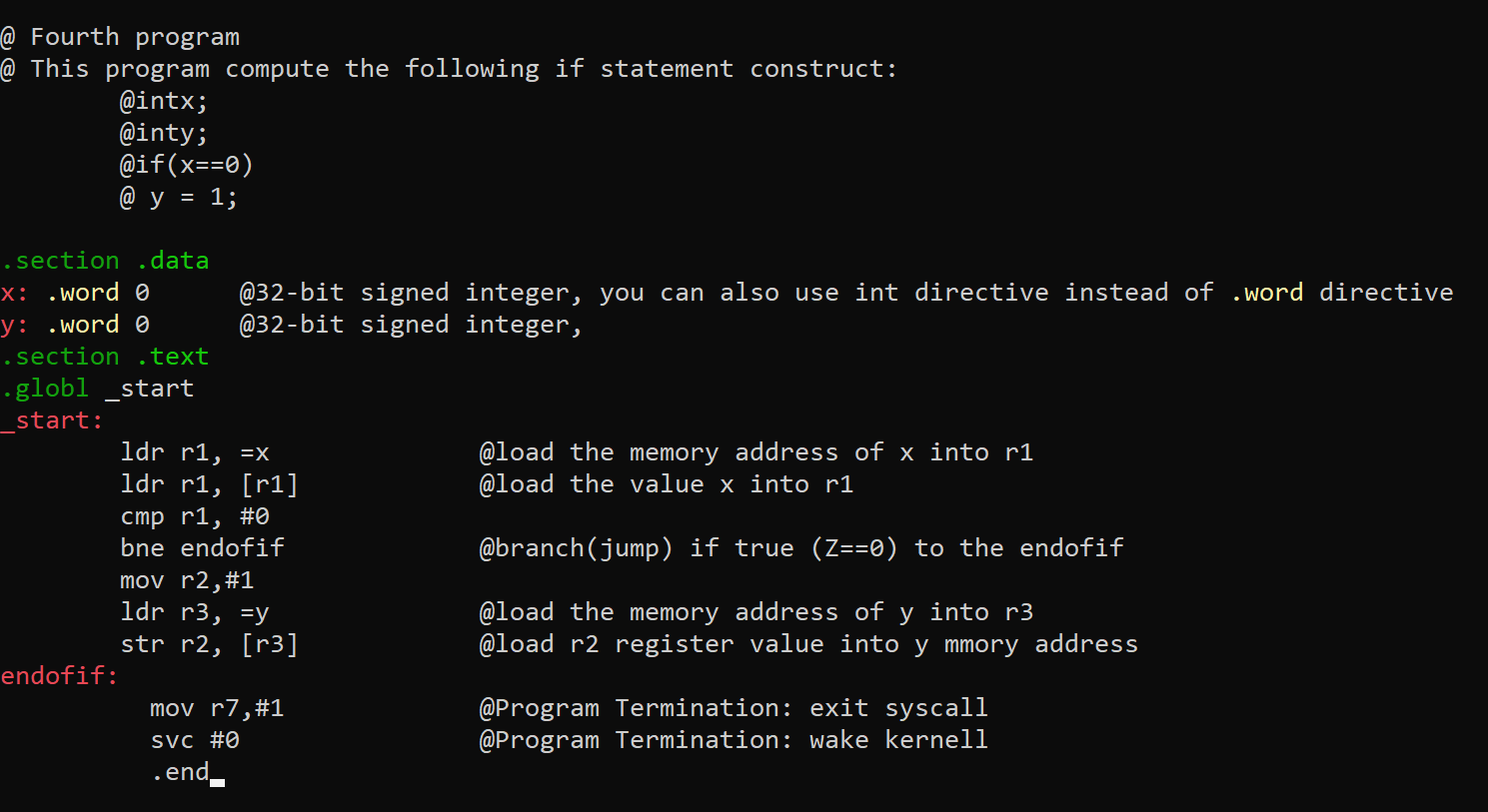
A close up of a logo

Description automatically generated

*Figure 7*

As shown above the zero and carry flags were set.

Part b



*Figure 8*

As shown above I replaced beq with bne to make the program more efficient as shown in the instructions.

A breakpoint was then put in line 15 and the registers were checked as shown in the figure below.

A screenshot of a cell phone

Description automatically generated

*Figure 9*

The value for x is shown above in Figure 9 and the value for y is shown below in figure 10

A screenshot of a cell phone

Description automatically generated

*Figure 10*

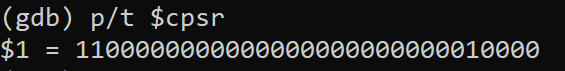
Then as shown below the value of 1 is assigned to y at the end of the program as The program did not jump to the endofif part of the program.

A screenshot of a cell phone

Description automatically generated

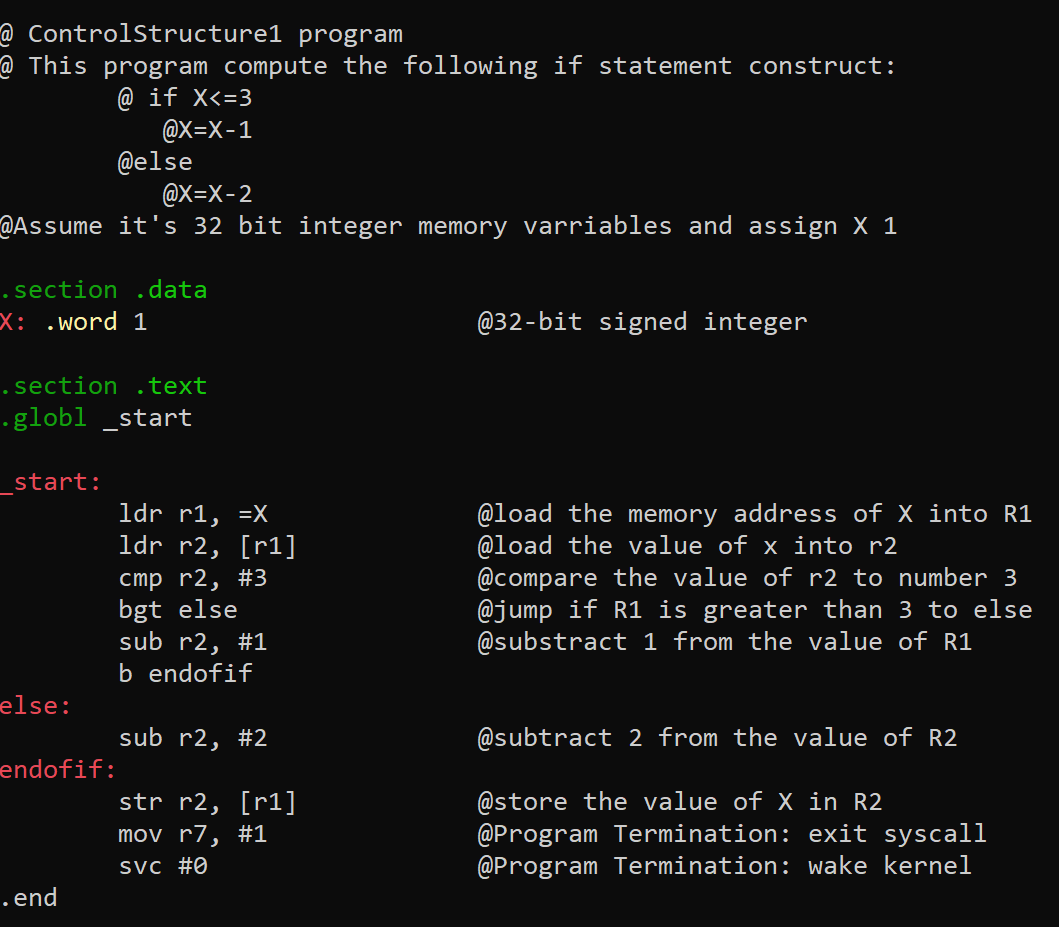
*Figure 11*

The flags are shown below and they were the same as part a



*Figure 12*

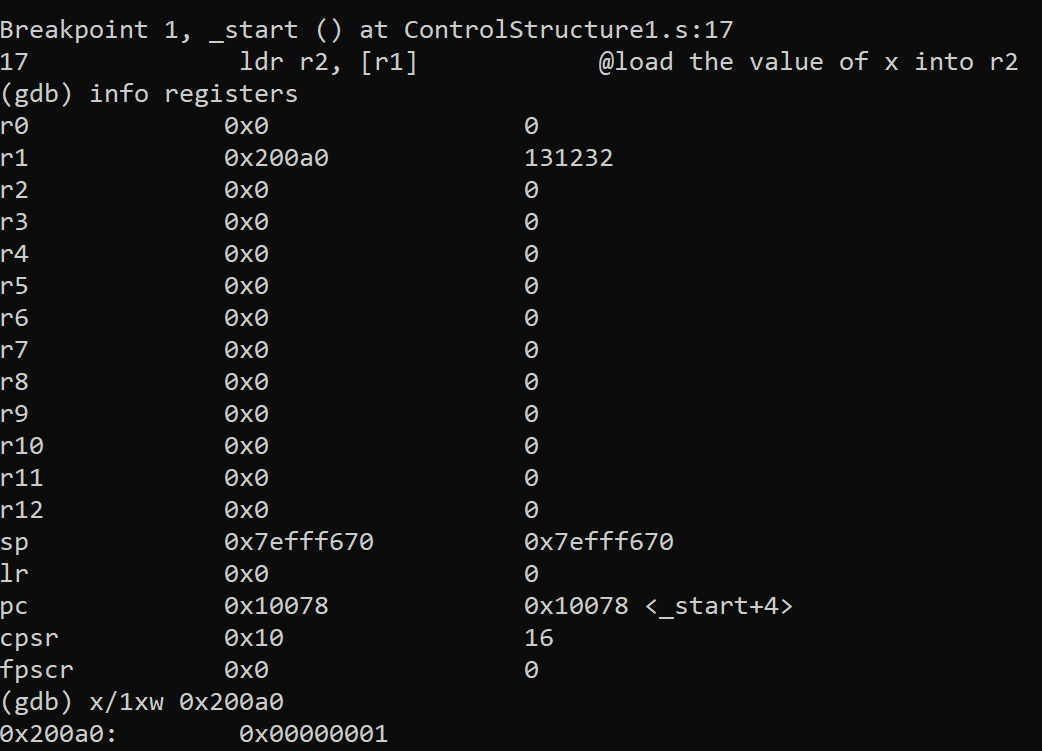
Part c



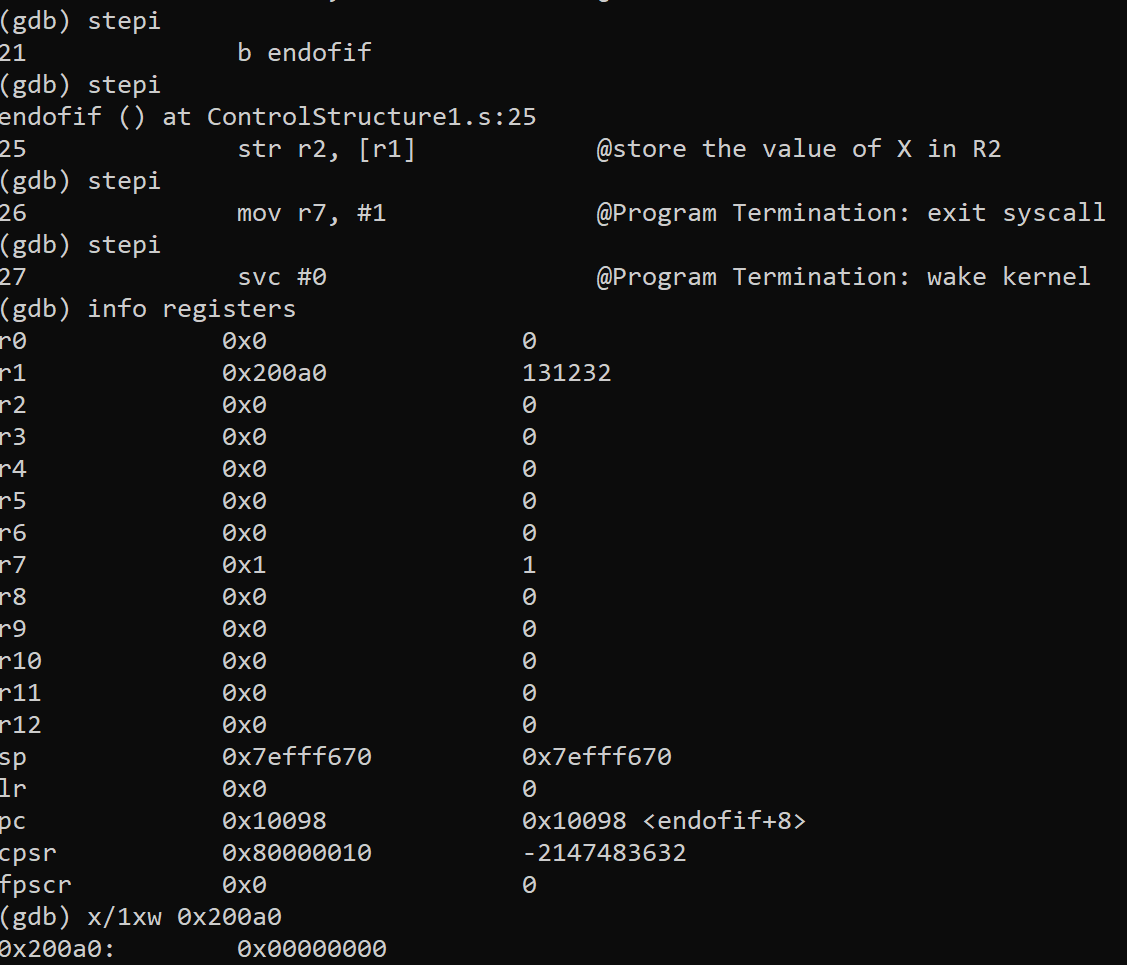
*Figure 13*

The ControlStructure1 program is shown above in Figure 13

The Debugging is shown below and the value of x is shown

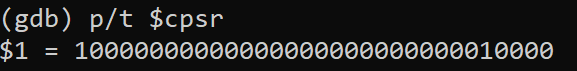


*Figure 14*

**

*Figure 15*

The debugging is shown in Figure 15. The result is 0. The Flags are shown in Figure 16

**

*Figure 16*