



09/04/2024

F TO C REPORT

Prepared For :
Dr.lamiaa

Description of the Problem:

The problem requires writing a MIPS assembly program to convert Fahrenheit temperatures to Celsius. The formula to perform this conversion is:

$$\text{Celsius} = (\text{Fahrenheit} - 32) * (5/9)$$

Sample Input:

The program expects the user to input a floating-point value representing the temperature in Fahrenheit like 85

Sample Output:

Upon receiving the input Fahrenheit temperature, the program calculates the equivalent temperature in Celsius and outputs it : 29.444447

Problem Statement:

The program should prompt the user to input a temperature value in Fahrenheit. Upon receiving the input, it should calculate the equivalent temperature in Celsius using the provided formula. Finally, the program should display the converted temperature in Celsius to the user.

Updates to Make the Example Work:

1. Comments for Improved Understanding:

Throughout the code, comments are added to explain the purpose and functionality of each section. This includes comments for input/output operations, procedure calls, and mathematical operations involved in the Fahrenheit to Celsius conversion.

2. Definition of Constants in .data Section:

Constants such as `const9`, `const5`, and `const32` are declared in the `.data` section, each with their corresponding floating-point values. This approach provides clarity and allows easy modification of constants if needed.

3. Corrected Syntax for Loading Constants:

The original code used the `lwc1` instruction to load constants into floating-point registers. However, the correct instruction for loading floating-point constants is `l.s`. This correction ensures the proper loading of constants into registers.

4. Proper Passing of Arguments:

Before calling the `f2c` procedure, the Fahrenheit temperature input by the user is moved from `$f0` to `$f12`, the designated argument register. This ensures that the procedure receives the correct input for conversion.

6. Ensured Proper Program Termination:

To guarantee the correct termination of the program, the code sets $\$v0$ to 10 before invoking a syscall. This syscall signals the termination of the program, ensuring a graceful exit.

C:\Users\MF\Desktop\assignment (2)\F to c\F to Casm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

Text Segment

Byte	Address	Code	Basic	Source
	4194304	0x24020004	addiu \$2,\$0,4	11: li \$v0,4
	4194308	0x3c011001	lui \$1,4097	12: la \$a0,input
	4194312	0x34240000	ori \$4,\$1,0	
	4194316	0x0000000c	syscall	13: syscall
	4194320	0x24020006	addiu \$2,\$0,6	15: li \$v0,6
	4194324	0x0000000c	syscall	16: syscall
	4194328	0x24020004	addiu \$2,\$0,4	19: li \$v0,4
	4194332	0x3c011001	lui \$1,4097	20: la \$a0,output
	4194336	0x34240012	ori \$4,\$1,18	
	4194340	0x0000000c	syscall	21: syscall
	4194344	0x4000030e	mov.s \$f12,\$f0	24: mov.s \$f12,\$f0
	4194348	0x0c100011	jal 4194372	25: jal \$f0
	4194352	0x24020002	addiu \$2,\$0,2	28: li \$v0,2
	4194356	0x4000030e	mov.s \$f12,\$f0	29: mov.s \$f12,\$f0
	4194360	0x0000000c	syscall	30: syscall
	4194364	0x2402000a	addiu \$2,\$0,10	33: li \$v0,10
	4194368	0x0000000c	syscall	34: syscall

Labels

Label	Address
(global)	
main	4194304
F to C.asm	4194372
\$2c	268500992
input	268501010
output	268501028
const9	268501032
const5	268501036
const32	268501036

☒ Data ☒ Text

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+12)	Value (+16)	Value (+20)	Value (+24)	Value (+28)
268500992	1702129253	1752440946	1635131493	543520108	1752432650	1970217061	1953853556	544434464
268501024	10	1091567616	1084227594	1107296256	0	0	0	0
268501056	0	0	0	0	0	0	0	0
268501088	0	0	0	0	0	0	0	0
268501120	0	0	0	0	0	0	0	0
268501152	0	0	0	0	0	0	0	0
268501184	0	0	0	0	0	0	0	0
268501216	0	0	0	0	0	0	0	0
268501248	0	0	0	0	0	0	0	0
268501280	0	0	0	0	0	0	0	0
268501312	0	0	0	0	0	0	0	0
268501344	0	0	0	0	0	0	0	0
268501376	0	0	0	0	0	0	0	0
268501408	0	0	0	0	0	0	0	0

0x10010000 (data) ☐ Hexadecimal Addresses ☐ Hexadecimal Values ☐ ASCII

Mars Messages Run IO

enter the value
95
the output is
29.444447
-- program is finished running --

Clear

Registers Coproc 1 Coproc 0

Name	Number	Value
\$zero	0	0
\$at	1	268500992
\$v0	2	10
\$v1	3	0
\$a0	4	268501010
\$a1	5	0
\$a2	6	0
\$a3	7	0
\$t0	8	0
\$t1	9	0
\$t2	10	0
\$t3	11	0
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$s0	16	0
\$s1	17	0
\$s2	18	0
\$s3	19	0
\$s4	20	0
\$s5	21	0
\$s6	22	0
\$s7	23	0
\$t8	24	0
\$t9	25	0
\$k0	26	0
\$k1	27	0
\$w0	28	268468224
\$w1	29	2147479548
\$f0	30	0
\$f1	31	4194352
pc		4194372
hi		0
lo		0