CS 226

Computer Organization and Design

Fall 2024

Assignment #4

Data Definitions and Memory Allocation

Due date: Wednesday, September 18, 2024

As we have discussed in class, an important part of programming is working with data of various types. Defining data in assembly language programs occurs in the .data segment of the program using assembler directives. For this assignment, you will define a collection of data of various types using the appropriate assembler directives as was described in class.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Starting with the shell program we created in class, define the data values listed below as described in the order given. You will need to determine the appropriate assembler directive to use in defining each item. For numeric values, use the smallest allocation which will be adequate to correctly define the data (.byte, .half, .word).

Every data item must be assigned a user defined symbol name, synonymous with a variable name in high level language programming. Symbol names must begin with an alphabetic character, and may contain numeric digits and an underscore. Shorter symbol names (5 – 10 characters) are preferred.

1. A number in the range -128 to +127.

2. A character string containing 15 alphanumeric characters.

3. A number in the range -32768 to +32767.

4. The uppercase alphabetic character Y.

5. The number 1 million.

6. An array containing the following numbers: 1 19 52 79 12

7. An allocation of empty memory space equivalent to the total

number of bytes required for data items 1 through 6, including any

non-allocated memory “holes”.

Save your program, naming it according to the following file name format:

Lastname\_HW4.asm. Don’t leave off the .asm filename extension.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Assemble your program to make sure you have no syntax errors. It is not necessary to run the program since no additional executable code has been added to the program. The .text segment should remain unchanged from the shell program format. You should edit the header comments to include a short description of your program based on the assignment.

­­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

On the last page of this assignment, you will find a memory diagram similar to that presented as the logical view of memory in the lecture materials (Chapter 2 Part 2). Using this diagram, indicate how the above data definitions will be allocated in memory as was shown in the example in the lecture slides. Be sure to indicate any non-allocated memory “holes” using the symbol ⊗.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submit on Blackboard a copy of your assembly language source code program file and a copy of your completed memory diagram.

Grading Criteria

Appropriate header comments 2 points

Correct data definitions in .data segment 7 points

Appropriate indentation and vertical alignment 2 points

Completed memory diagram based on your definitions 7 points

Total 18 points

