Dept. of CSE, Bennett University

ECSE217L – Microprocessor and Computer Architecture

Lab Assignment – 8

You will learn how to use array in MIPS to solve problems in this lab. Remember that each integer on the MIPS architecture requires 4 bytes to declare an array of integer-sized components (or 32 bits). In addition, with the MIPS architecture, each word is 4 bytes long. As a result, we can declare an array of integers using the .word directive:

```
int_array:  .word   0:36
```

This declaration allocates 36 words (integer-sized memory chunks), which are all (nicely) located at word-aligned addresses. The initial value of each array element is 0.

Array Declaration with Initialization:

```
.data
vowels: .byte 'a', 'e', 'i', 'o', 'u'
pow2: .word 1, 2, 4, 8, 16, 32, 64, 128
```

Vowels names a contiguous block of 5 bytes, set to store the given values; each value is stored in a single byte.

Address of vowels[k] == vowels + k

pow2 names a contiguous block of 32 bytes, set to store the given values; each value is stored in a word (4 bytes)

Address of pow2[k] == pow2 + 4 * k

This mechanism shows an agreement (also called a protocol) between the caller who provides the arguments and callee who uses the arguments to do computations and return a value, and finally back to the caller who uses the return value.

Now write the following programs in MIPS.

1. Write a programme to calculate the sum of all the numbers in a certain range. Take the numbers from the user, save them in an Array, then add them up.

   Example: Enter the first number: 1
   Enter the second number: 5
   The sum is: 15 ## 1+2+3+4+5 = 15

2. Write a program to print an array in reverse order. First store sum numbers in an Array, take the numbers as user input. Then print the array with the number in reverse order.

   Example:
   ```
   Enter the value in Array:
   0
   Enter the value in Array:
   1
   Enter the value in Array:
   2
   …
   …
   Enter the value in Array:
   9

   Entered Array: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
   Reverse Order: 9, 8, 7, 6, 5, 4, 3, 2, 1, 0
   ```

3. Using MIPS programming, create a programme to implement Bubble sort.

   Example: The elements in the array: 1 5 4 3 2

   The sorted array is: 1 2 3 4 5

4. Declare an array in RAM having 3 elements each initialized to 100

   Example:
   ```
   100
   100
   100
   ```

Submission Instructions:

- Submit your .asm files from LMS within 4 days from the day of your lab slot. Save all the files as per the format **RollNo_Lab#_QuestionNo.asm (Example: E19CSE632_Lab8_Q2.asm)**.
- Write your Name and Roll No. as comment before starting of each program.
- Make it sure that in each program, you have mentioned enough comments regarding the explanation of program instructions.
- In the LMS please submit in your respective batch's submission portal. Submission in other batch's submission portal will not be checked.

- Write your Name and Roll No in the .m file itself (Use # to insert comment lines). Without this you will score zero for that particular question.
- Late submission will lead to penalty.
- Any form of plagiarism/copying from peer or internet sources will lead penalty.