

Assembly Language Programming D
FAST-NU, Lahore, Spring 2018

Homework 3

Due Wednesday February 28 11:55 P.M.

Marked out of 50 points.

Write code for the following problems. Submit soft copies via SLATE.

1) Write a program that packs a set of given numbers into a 'bit vector'. The original numbers are from the range 0 and 1023 and stored in an array called arr1 of a given size. Your program should maintain another array called bvec of exactly 1024 bits that are all initially zero; the left most bit is numbered 0 and the right most bit is numbered 1023. The program should go through the array arr1 and for each word k in this array it should set the kth bit of bvec.

2) AX contains a non-zero number. Count the number of ones in it and store the result back in AX. Repeat the process on the result (AX) until AX contains one. Calculate in BX the number of iterations it took to make AX one. For example BX should contain 2 in the following case:

AX = 1100 0101 1010 0011 (input – 8 ones)

AX = 0000 0000 0000 1000 (after first iteration – 1 one)

AX = 0000 0000 0000 0001 (after second iteration – 1 one) STOP

3) Your program contains an array arr1 of a given size and random data. Your program goes through the array and if a word contains an even number it should replace the number with the number of zeroes in the word, else, if the word contains an odd number it should be replaced with the number of ones in the word.

4) Recall from class the program that shifted an entire array, arr1, of a given size, left by one place. Now add more functionality to that program, so that it can rotate the bits of the array to left by a given number of places; this number is stored in a variable called count. Assume that the value of count lies between 0 and 15.

THE END