W

rite a MIPS program implementation for the C-like pseudo code given in the accompanying ***DynamicArrays.s*** file. The given algorithm converts a counting ***number*** to a string representation. Without loss of generality, the value of ***number*** is hard coded.

|  |
| --- |
| **# int number = 0x7FFF**  **# string reversed**  **# string inOrder**  **# int length = 0**  **# int r = 0**  **# int k = 0**  **# char ch**  **# string label1 = " Digits reversed = "**  **# string label2 = " Digits in order = "**  **# void main()**  **# {**  **# reversed = new char[ 16 ]**  **# do**  **# {**  **# ch = (char) ( number % 10 + (int) '0' )**  **# reversed[ r ] = ch**  **# r++**  **# number = number / 10**  **# }**  **# while (number > 0)**  **# reversed[ r ] = chr(0)**  **# length = r**  **# inOrder = new char[ 16 ]**  **# r = length - 1**  **# while ( r >= 0 )**  **# {**  **# inOrder[ k ] = reversed[ r ]**  **# k++**  **# r--**  **# }**  **# digitsInOrder[ length ] = chr(0)**  **# Console.Write( label1 )**  **# Console.Write( reversed )**  **# Console.Write( label2 )**  **# Console.Write( inOrder )**  **# }** |

O

rganize your MIPS program to satisfy our MIPS Program Coding Requirements. Save your source code for this program back to the same file, ***DynamicArrays.s***.

A

t the end of a successful execution, save to file a copy of the Data segment and also the HEAP. Then create a Microsoft Word document, *DynamicArrays.doc* (with landscape orientation selected in page setup) that contains

1. The project title, **Integer to String**
2. The MIPS program **source code** -- New Courier Font, size 10, left and right margins = 0.5
3. A copy of the **Data segment**. Use the draw facility of Microsoft Word to label the stored values for the nine variables.

H

and in a printed copy of *DynamicArrays.doc*, and also submit this file on Blackboard for the Dynamic Arrays (DA) assignment in the Assignment folder.