JOINT INSTITUTE
交大密西根学院

ECE3700J Introduction to Computer Organization
Lab 2 Assembly Programming

Purpose

This lab is design for you to have a programming experience with RISC-V assembly instruction set.

Tasks

Develop a RISC-V assembly program that operates on a data segment consisting of an array of 32-bit signed integers. In the text (program) segment of memory, write a procedure called `main` that implements the `main()` function as well as procedures for other subroutines described below. Assemble, simulate, and carefully comment the file. Screen print your simulation results and explain the results by annotating the screen prints. You should compose an array whose size is determined by you in the `main` function and is not less than 30 elements.

In this lab, you are allowed to use RV32I BASE INTEGER INSTRUCTIONS. Any Extension Instruction Sets or pseudo-instructions are not allowed.

```
main() {
  int size = ...; //determine the size of the array here
  int PosCnt, NegCnt, ZeroCnt, Sum;
  int testArray[size] = { 1, 13, 0, -56,
                         ... //compose your own array here
                             //test data will be provided at demonstration
                        };

  //PosCnt should hold the number of positive values in testArray[]
  PosCnt = countArray(______, ______, _______);

  //NegCnt should hold the number of negative values in testArray[]
  NegCnt = countArray(______, ______, _______);

  //ZeroCnt should hold the number of zeros in testArray[]
  ZeroCnt = countArray(______, ______, _______);

  //Sum should hold the sum of all values in testArray[]
  Sum = sumArray(______, ______);
}

/***********************************************************************
 * countArray(int A[], int numElements, int cntType);
 * Count specific elements in the integer array A[] whose size is     *
 * numElements and return the following:                              *
 * When cntType = 1, count the positive elements;                     *
 * When cntType = -1, count the negative elements;                    *
 * When cntType = 0, count the elements that are zeros.               *
 ***********************************************************************/
int countArray(int A[], int numElements, int cntType) {
  int cnt = 0;

  ...... //complete the code here
         //must call functions isPos(), isNeg(), and isZero()
  return cnt;
}
```

```
int isPos(int x) {
  if(x>0) return 1;
  else return 0;
}

int isNeg(int x) {
  if (x<0) return 1;
  else return 0;
}

int isZero(int x) {
  if (x==0) return 1;
  else return 0;
}

/*************************************************************
 * sumArray(...) returns the sum of all elements in an array *
 *************************************************************/
int sumArray(______, ______){

  ...... //complete the function here

}
```

Deliverables

This is a 2-week lab. The full score for this lab is 200 points.

1) Demonstrate your program and simulation results to the TAs before your lab session ends. Go through the program step by step and show corresponding changes in the registers and memory.
2) Upload the source file(s) on Canvas by **22:00 pm, October 8, 2022**.

This is an individual assignment. Your work must be submitted electronically to Canvas before the specified due date. Late submission will result in 0 point for the corresponding deliverables. Source code must be submitted before a grade for this lab can be assigned.

Grading Policy

Demonstration: 80%
Source code: 20%