**CS2100 Computer Organization**

**Assignment 1**

**Answer Book**

Please see instructions in the main assignment sheet. You must convert this file to PDF and name it AxxxxxxY.pdf before submission. AxxxxxxY is your student ID. **-3 marks for improperly named submissions.**

Question 0 **(-3 marks if missing**)

|  |  |
| --- | --- |
| **Name:** | Izz Hafeez Bin Zek Hazley |
| **Student ID:** | A0234693X |
| **Tutorial Group:** | 21 |

Question 1 (15 MARKS)

1a. (0.5 marks)

Size of each array element: 8 bits

1b. (2.5 marks)

Elements of t in binary:

0b00110011, 0b11100011, 0b00111111, 0b11110111, 0b11111111

1c. (8 marks)

|  |  |
| --- | --- |
| **Variable** | **Purpose (in one short sentence)** |
| a | Keep track of the number of consecutive 1s in x |
| b | Index to loop through the elements of x |
| r | Keep track of the maximum length of consecutive 1s in x |
| d | Index to loop through the bits of c |

1d. (4 marks)

Describe in **under 20 words** what the mystery function does:

Finds the length of the longest substring of 1s in the binary representation of all elements in x

Question 2 (5 marks)

// buffer = buffer of char

// len = # of char in buffer

// odd = 1 set bit 7 of each item of buffer according to rules of odd parity.

// Otherwise set according to the rules of even parity.

void set\_parity(char \*buffer, int len, int odd) {

for (int i=0; i<len; i++) {

char c = buffer[i];

char c\_cpy = c; // Gets rid of the first digit

int count = 0;

int j = 0;

while (j < 8) {

if (c\_cpy & MASK) { // MSB is 1

count++;

}

c\_cpy <<= 1;

j++;

}

printf("%d\n", count);

if (((count % 2) && !odd) || (!(count % 2) && odd)) {

buffer[i] = c | MASK;

} else {

buffer[i] = c;

}

}

}

Question 3 (10 MARKS)

Part a (9 marks)

|  |  |
| --- | --- |
| Variable Mappings | Comments |
| address of array “A[]” 🡺 $s0  i 🡺 $s1 (**initialized to 5**)  j 🡺 $s2 | |
| **addi $s2, $zero, 1** | j = 1 |
| **loop:**  **slt $t8, $s1, $s2**  **bne $t8, $zero, end** | while (j <= i) { |
| **sll $t3, $s2, 2**  **add $t5, $s0, $t3**  **subi $t4, $t5, 4**  **lw $t0, 0($t4)** | *t0* = A[j-1] |
| **sll $t3, $s2, 2**  **add $t5, $s0, $t3**  **lw $t1, 0($t5)** | *t1* = A[j] |
| **slt $t7, $t1, $t0**  **beq $t7, $zero, skip** | if (a[j-1] > a[j] ) { |
| **sw $t0, 0($t5)**  **sw $t1, 0($t4)** | //swap a[j-1] with a[j]  }//end of if |
| **skip:**  **addi $s2, $s2, 1**  **j loop** | j++  } //end of while |
| **end:** |  |

Part b (1 mark)

**sll $t3, $s2, 2 = 0x00125880**

**000000 00000 10010 01011 00010 000000**

Question 4 (10 MARKS)

0.5 marks each

|  |  |  |  |
| --- | --- | --- | --- |
| RegDst | 0 | ALUOp |  |
| RegWrite | 0 | isZero |  |
| ALUSrc | 1 | Sign Extend Output\* | 0x |
| PCSrc | 0 | **A\*** | 0x |
| ALUControl | 0000 | **B** |  |
| ALUResult\* | 0x | Inst[31-26]\* | 0b |
| MemRead |  | Inst[25:21]\* | 0b |
| MemWrite |  | Inst[20:16]\* | 0b |
| MemToReg |  | Inst[15:11]\* | 0b |
| WriteData\* | 0x | Inst[5:0]\* | 0b |