Course Name: _____

Course Number and Section: **14:332:xxx:xx**

Experiment: [Experiment # [3] – C Memory Management and Introduction to RISC-V]

Lab Instructor:

Date Performed:

Date Submitted:

Submitted by: [Roshni Shah 172005723]

Course Name: _____

Course Number and Section: **14:332:xxx:xx**

! Important: Please include this page in your report if the submission is a paper submission. For electronic submission (email or Sakai) please omit this page.

----------For Lab Instructor Use ONLY----------

GRADE: _____

COMMENTS:

Electrical and Computer Engineering Department
School of Engineering
Rutgers University, Piscataway, NJ 08854

Question 1:

1) Static Variables --- static
2) Local Variables --- stack
3) Global Variables --- static
4) Constants --- Code, Static, Stack
5) Machine Instructions --- code
6) maaloc() --- heap
7) String Literals --- Static

Question 2:

a) arr= (int *) malloc(sizeof(int)* k);
b) str = (char *) malloc(sizeof(char) * (p + 1));
c) mat = (int **) calloc(n, sizeof(int *));
for (int i = 0; i < m; i++) {
mat[i] = (int *) calloc(m, sizeof(int));
}

Question 3:

a) sets the register t0 to have the value arr[3] which was in register s0
b) Increments the array element as mentioned by t2 (such as arr[t2]) by 1
c) The register t0 is set to the two's complement negation of arr[0]

Question 4:

s0<s1	s0<= s1	s0>1
slt t0, s0, s1 bne t0, 0, label	slt t0, s1, s0 beq t0, 0, label	sltiu t0, s0, 2 beq t0, 0, label

Question 5:

In the code file

Question 6:

C	RISC-V
// s0 -> a, s1 -> b **//s2 -> c, s3 -> z** **int a = 4, b = 5, c = 6, z; z = a + b + c + 10;**	**addi s0, x0, 4** **addi s1, x0, 5** **addi s2, x0, 6** **add s3, s0, s1** **add s3, s3, s2** **addi s3, s3, 10**
// s0 -> int * p = intArr;	**sw x0, 0(s0)**

// s1 -> a; *p = 0;
int a = 2;
p[1] = p[a] = a;
	addi s1, x0, 2
	sw s1, 4(s0)
	slli t0, s1, 2
	add t0, t0, s0
	sw s1, 0(t0)

// s0 -> a, s1 -> b
int a = 5, b = 10;
if(a + a == b) {
a = 0;
} else {
b = a - 1;
}
	addi s0, x0, 5
	addi s1, x0, 10
	add t0, s0, s0
	bne t0, s1, else
	xor s0, x0, x0
	jal x0, exit
	else: addi s1, s0, -1
	exit:

s1 = 1;
for(s0=0;s0<30;s++) {
s1 *= 2;
}
	addi s0, x0, 0
	addi s1, x0, 1
	addi t0, x0, 30
	loop:
	beq s0, t0, exit
	add s1, s1, s1
	addi s0, s0, 1 jal x0, loop
	exit:

// s0 -> n, s1 -> sum
// assume n > 0 to start
int sum;
for(sum=0;n>0;sum+=n--);
	addi s1, s1, 0
	loop:
	beq s0, x0, exit
	add s1, s1, s0
	add s0, s0, -1
	jal x0, loop
	exit:

Question 7:

In code file