MIPS cheat sheet

Instruction	Syntax	Example
add	add dest, src0, src1	add $s0, $s1, $s2
sub	sub dest, src0, src1	sub $s0, $s1, $s2
addi	addi dest, src0, immediate	addi $s0, $s1, 12
lw	lw dest, offset(base addr)	lw $t0, 4($s0)
sw	sw src, offset(base addr)	sw $t0, 4($s0)
bne	bne src0, src1, branchAddr	bne $t0, $t1, notEq
beq	beq src0, src1, branchAddr	bne $t0, $t1, Eq
j	j jumpAddr	j jumpWhenDone

C	MIPS

`// $s0 -> a, $s1 -> b`
`// $s2 -> c, $s3 -> z`
 	
`int a=4, b=5, c=6, z;`
`z = a+b+c+10;`	

`// $s0 -> int *p = (int *)malloc`
`//              (3*sizeof(int));`
`// $s1 -> a`
`p[0] = 0;`
`int a = 2;`
`p[1] = a;`
`p[a] = a;`	

`// $s0 -> a, $s1 -> b`
`int a = 5, b = 10;`
`if (a + a == b) {`
`    a = 0;`
`} else {`
`    b = a - 1;`
`}`	

`/*What does this do?`
`  (Not C, in English) */`
	`      addi $s0, $0,  0`
	`      addi $s1, $0,  1`
	`      addi $t0, $0,  30`
	`loop: beq  $s0, $t0, done`
	`      add  $s1, $s1, $s1`
	`      addi $s0, $s0, 1`
	`      j    loop`
	`done: # done!`

Implement `streq`, which sets `$v0` to true if its two character pointer arguments (`$a0` and `$a1`) point to equal strings (and false otherwise), in MIPS (there is not enough room to do this on this sheet of paper; use another).

What are the instructions to branch on each of the following conditions?

`$s0 < $s1`

`$s0 <= $s1`

`$s0 > 1`

`$s0 >= 1`

What are the 3 meanings unsigned can have in MIPS?

Pointer Exercises

1) Given the following definition of a binary tree, implement a binary search routine for the tree.

```
struct btree {              // return true iff x is in the tree.
  int value;                // Assume left branch is "less".
  struct btree              int bsearch(struct btree *t, int x) {
    *left, *right;
};




                            }
```

2) What will C print? Assume the following. The `%p` format specifier prints a pointer.

`int arr[9]; // lives at address 0x10000000`

`printf("%p\n",arr);`	`printf("%p\n",arr+1);`
`printf("%p\n",&arr[0]);`	`printf("%p\n",&arr[0]+1);`