Conditionals, Loops

Agenda

- Conditionals and Loops Examples
- Lab 4 Overview
- Project 1 overview

Instruction	Meaning
`beq a, b, label`	`if(a == b) { goto label }`
`bne a, b, label`	`if(a != b) { goto label }`

Instruction	Meaning
`bltz a, label`	`if(a < 0) { goto label }`
`blez a, label`	`if(a <= 0) { goto label }`
`bgtz a, label`	`if(a > 0) { goto label }`
`bgez a, label`	`if(a >= 0) { goto label }`

https://luisfnqoliveira.gitlab.io/cs447_s2021/slides/07-Flow,-Conditionals,-and-Loops.pdf

Instruction	Meaning
`slt c, a, b`	`if(a < b) { c = 1 } else { c = 0 }`

Set if Less Than: register c will be set to 1 if a<b.
Otherwise, register c will be set to 0.

Instruction	Meaning
`blt a, b, label`	`if(a < b) { goto label }`
`ble a, b, label`	`if(a <= b) { goto label }`
`bgt a, b, label`	`if(a > b) { goto label }`
`bge a, b, label`	`if(a >= b) { goto label }`

https://luisfnqoliveira.gitlab.io/cs447_s2021/slides/07-Flow,-Conditionals,-and-Loops.pdf

Exercise – Convert from C to MIPS

```
if (i == j)
{
        i++;
}
else
{
        j--;
}
j += i;
```

s1	i
s2	j

MIPS Code

```
bne s1, s2, ELSE

addi s1, s1, 1 # i++

j NEXT # jump over else

ELSE: addi s2, s2, -1 # else j--

NEXT: add s2, s2, s1 # j += i
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

Exercise 2

```
if(i < j)
{ i++;}
if(i > 100)
{ i --; }
else
{ i ++;}
i+=j;
```

s1	i
s2	j

Pseudo Code

```
if i >= j goto condition2
        i ++
condition2: if i <= 100 goto Else
        i - -
        jump to NEXT
Else: i ++
NEXT: i + = j
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

Exercise 2

Pseudo Code

```
if i >= j goto condition2
        i ++
condition2: if i <= 100 goto Else
        i - -
        jump to NEXT
Else: i ++
NEXT: i + = j
```

MIPS Code

```
ble s1, s2,cond2
addi s1, s1, 1 # i++

cond2: ble i, 100, Else
addi s1, s1, -1 # i--
j Next

Else: addi s1, s1, 1 # i++

Next: add s1, s2, s1
```

Loops - While

```
while ( <cond> ) {
    <while-body>
}
```

```
L1: if ( <cond> ) {
        <while-body>
        goto L1 ;
    }
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

Example

```
while (i < j)
{
    k++;
    i = i * 2;
}
```

Pseudo Code 1

```
L1: if (i < j)
{
    k ++
    i = i *2
    goto L1
}
```

Example

Pseudo Code 1

```
L1: if (i < j)
{
      k ++
      i = i *2
      goto L1
}
```

Pseudo Code 2

```
L1: if (i >= j) goto OutsideLoop
      k ++
      i = i *2
      Jump to L1
OutsideLoop
```

Example

Pseudo Code 2

```
L1: if (i >= j) goto
OutsideLoop
      k ++
      i = i *2
      Jump to L1
OutsideLoop
```

MIPS

```
L1: bge i, j, OutsideLoop
      addi s3, s3, 1
      add s1, s1, s1
      j L1
OutsideLoop:
```

Loops - For

```
for ( <init> ; <cond> ; <update> ) {
      <for-body>
}
```

```
<init>;
while ( <cond> ) {
      <for-body>
      <update>
}
```

```
      <init>;
L1:   if ( <cond> ) {
            <for-body>
            <update>
            goto L1 ;
      }
DONE:
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

Example

```
main () {
        int i, size = 10, sum, pos, neg;
        int arr[10] = {12, -1, 8, 0, 6, 85, -74, 23, 99, -30};

        sum = 0; pos = 0; neg = 0;
        for (i = 0; i < size; i++) {
                sum += arr[i];
                if (arr[i] > 0)
                        pos += arr[i];
                if (arr[i] < 0)
                        neg += arr[i];
        }
        return 0;
}
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

Example

```
.data
        size .word 10
        arr .word 12, -1, 8, 0, 6, 85, -74, 23, 99, -30
.text
.globl main
main:
#1. Load each of the variables to a register and initialize them
    la s0, size         # initialize registers
    lw s1, 0(s0)        # $s1 = size
    ori s2, zero, 0     # $s2 = sum
    ori s3, zero, 0     # $s3 = pos
    ori s4, zero, 0     # $s4 = neg
# <init>
    ori s5, zero, 0     # $s5 = i
    la s6, arr          # $s6 = &arr
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

Example

```
# if (<cond>)
L1: bge s5, s1, DONE
# <for-body>
        lw s7, 0(s6)          # $s7 = arr[i]
        addu s2, s2, s7       # sum += arr[i]
        blez s7, NEG          # if ! (arr[i] > 0)
        addu s3, s3, s7       # pos += arr[i];

        j UPDATE              # goto UPDATE
NEG:    bgez $s7, UPDATE      # if ! (arr[i] < 0)
        addu $s4, $s4, $s7    # neg += arr[i];
UPDATE:                       # <update>
        addi $s5, $s5, 1      # i++
        addi $s6, $s6, 4      # move array pointer
        j L1                  # goto L1
DONE:
```

https://www.cs.umd.edu/~meesh/cmsc311/clin-cmsc311/Lectures/lecture15/C_code.pdf

MIPS to C

MIPS Code

```
ble s1, s2,cond2
addi s1, s1, 1 # i++

cond2: ble s1, 100, Else
addi s1, s1, -1 # i--
j Next

Else: addi s1, s1, 1 # i++

Next: add s1, s2, s1
```

Pseudo Code

```
if s1 <= s2 goto cond2
     s1 = s1 + 1
cond2: if s1 <= 100 goto Else

     s1 = s1 - 1
     jump Next
Else
     s1 = s1 + 1
Next
     s1 = s1 + s2
```

MIPS to C

Pseudo Code

```
if s1 <= s2 goto cond2
      s1 = s1 + 1
cond2: if s1 <= 100 goto Else

      s1 = s1 - 1
      jump Next
Else
      s1 = s1 + 1
Next
      s1 = s1 + s2
```

C Code

```
if(s1 > s2)
{
        s1++;
}
if(s1 > 100)
{
        s1--;
}
else{
        s1++;
}
s1 = s1 + s2;
```

MIPS to C

C Code

```
if(s1 > s2)
{
        s1++;
}
if(s1 > 100)
{
        s1--;
}
else{
        s1++;
}
s1 = s1 + s2;
```

C Code

```
if(i > j)
{
        i++;
}
if(i > 100)
{
        i --;
}
else{
        i++;
}
i+=j;
```