

Decisions

Lecture 6 Section 2.7

Robb T. Koether

Hampden-Sydney College

Fri, Sep 6, 2019

- 1 Decision Structures in C
- 2 Conditional Branches in MIPS
- 3 One-Way Decisions
- 4 Two-Way Decisions
- 5 Assignment

Outline

- 1 Decision Structures in C
- 2 Conditional Branches in MIPS
- 3 One-Way Decisions
- 4 Two-Way Decisions
- 5 Assignment

Decision Structures in C

Decision Structures in C

```
if (condition)  
{  
    true-block  
}  
else  
{  
    false-block  
}
```

- The **if** statement is the basic decision structure in C.
- The **else** part is optional.

Outline

- 1 Decision Structures in C
- 2 Conditional Branches in MIPS**
- 3 One-Way Decisions
- 4 Two-Way Decisions
- 5 Assignment

Conditional Branches in MIPS

- **Conditional branches** are based on comparisons.
- Two quantities may be related in any of six ways:
 - Equal (eq)
 - Not equal (ne)
 - Less than (lt)
 - Greater than (gt)
 - Less than or equal to (le)
 - Greater than or equal to (ge)

Conditional Branches in MIPS

Instruction	Meaning
<code>beq <i>rs, rt, addr</i></code>	Branch if $rs == rt$
<code>bne <i>rs, rt, addr</i></code>	Branch if $rs \neq rt$
<code>bltz <i>rs, addr</i></code>	Branch if $rs < 0$
<code>blez <i>rs, addr</i></code>	Branch if $rs \leq 0$
<code>bgtz <i>rs, addr</i></code>	Branch if $rs > 0$
<code>bgez <i>rs, addr</i></code>	Branch if $rs \geq 0$

- MIPS provides a number of conditional branch instructions.
- Notice that the branch occurs if the condition is *satisfied*, which is the opposite of C.

Outline

- 1 Decision Structures in C
- 2 Conditional Branches in MIPS
- 3 One-Way Decisions**
- 4 Two-Way Decisions
- 5 Assignment

Example

Example (One-way Branch)

```
if (a == 10)
    a = 5;
```

- Translate the above C code into MIPS.
- Assume that `a` is stored in `$s0`.

Example

Example (Unstructured C)

```
if (a != 10) goto not10;  
a = 5;
```

```
not10:
```

- Write the code in unstructured C.
- Note that we branch if *a* does *not* equal 10.

Example

Example (One-way Branch)

```
li      $t0, 10           # Load 10
bne     $s0, $t0, not10  # Branch if a != 10
li      $s0, 5           # Set a = 5
not10:
```

- Assume that `a` is in `$s0`.

Example

Pseudo `bne` Instruction

```
bne      sr, imm, label # Branch if sr != imm
```

Example (One-way Branch)

```
bne      $s0, 10, not10  # Branch if a != 10  
li       $s0, 5          # Set a = 5  
not10:
```

- There is also a pseudo branch-not-equal instruction that takes an immediate value.

The Unconditional Branch

The Unconditional Branch

```
j          label          # Jump to label
```

- The jump instruction (`j`) is an **unconditional branch**.
- It will redirect execution to the statement with the label.
- The jump is necessary for a two-way decision structure.

Outline

- 1 Decision Structures in C
- 2 Conditional Branches in MIPS
- 3 One-Way Decisions
- 4 Two-Way Decisions**
- 5 Assignment

Example

Example (Two-way Branch)

```
if (a == b)
    c = d + e;
else
    c = d - e;
```

- Translate the above C code into MIPS.
- Assume
 - a is in \$s0
 - b is in \$s1
 - c is in \$t0
 - d is in \$s2
 - e is in \$s3.

Example

Example (Two-way Branch)

```
if (a != b) goto dec_else;  
c = d + e;  
goto dec_done;
```

```
dec_else:
```

```
c = d - e;
```

```
dec_done:
```

- Write the code in unstructured C.
- Note that we change the condition from `a == b` to `a != b`.

Example

Example (Two-way Branch)

```
    bne    $s0, $s1, dec_else # Branch if a != b
    add    $t0, $s2, $s3     # c = d + e
    j      dec_done
dec_else:
    sub    $t0, $s2, $s3     # c = d - e
dec_done:
```

Example

Example (Two-way Branch)

```
        beq      $s0, $s1, dec_else # Branch if a == b
        sub      $t0, $s2, $s3     # c = d - e
        j        dec_done
dec_else:
        add      $t0, $s2, $s3     # c = d + e
dec_done:
```

- Or we could keep the condition the same as in C, but reverse the order of the cases.

Examples

- Write a MIPS program that will assign a letter grade to a test score, using a 10-point scale.
- Write a MIPS program that will read three integers and display them in ascending order.

Outline

- 1 Decision Structures in C
- 2 Conditional Branches in MIPS
- 3 One-Way Decisions
- 4 Two-Way Decisions
- 5 Assignment**

Assignment

Assignment

- Read Section 2.7.