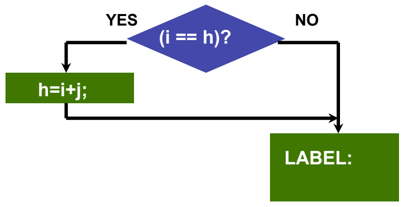
Control

- Instruction that potentially changes the flow of program execution
- MIPS conditional branch instructions
  - `bne $s4, $s3, LABEL` `goto LABEL if $s4 != $s3`
  - `beq $s4, $s3, LABEL` `goto LABEL if $s4 == $s3`
- Example

```
if (i == h) h =i+j;
```

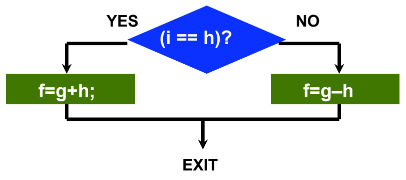


```
    bne $s5, $s6, LABEL
    add $s6, $s6, $s4
LABEL: ...
```

Control

- MIPS unconditional branch instruction (i.e., jump)
  - j LABEL
- Example
  - i, f, g, and h are in registers $s4, $s5, $s6, $s7

```
if (i == h) f=g+h;
else f=g–h;
```



```
    bne $s4, $s7, ELSE
    add $s5, $s6, $s7
    j EXIT
ELSE:
    sub $s5, $s6, $s7
EXIT:
    …
```

Control

- We have beq and bne; what about branch-if-less-than?
  - We have slt

```
if (s1<s2) t0=1;
else t0=0;
```

```
slt $t0, $s1, $s2
```

- Can you make a “psuedo” instruction “blt $s1, $s2, LABEL”?
- Assembler needs a temporary register to do this
  - $at is reserved for this purpose

```
slt     $at,$s1,$s2         # $at==1 when $s1<$s2
bne     $at,$0,LABEL        # $at!=0 implies $at==1
```

Address in I-format

Branch/ Immediate	op	rs	rt	16-bit immediate

- Immediate address is not a 32-bit value – it’s only 16-bit!
  - The 16-bit immediate value is signed (2’s complement form)
- Addressing in branch instructions
  - The 16-bit number in the instruction specifies the number of “instructions” to be skipped
  - Memory address is obtained by adding this number to the PC
  - Next address = PC + 4 + sign-extend(16-bit immediate << 2)
  - B/C 16-bit immediate is signed: can go both FORWARD and BACKWARD
- Example
  - beq $s1, $s2, 100

4	17	18	25

Example of Addresses in Branches

- Branch address is "PC relative"
  - The target is RELATIVE to address of the branch instruction itself
  - Next PC = PC + 4 + sign-extend(16-bit immediate << 2)

```
0x00400000              beq    $s0,$s1,LABEL  # what's LABEL???
0x00400004              andi   $s0,$s0,0xFF
0x00400008              srl    $s0,$s0,2
0x0040000c              lui    $s1,0xFF00
0x00400010              ori    $s1,$s0,$s1
0x00400014      LABEL:  …
   PC when branch taken=0x00400000 + 4 + sign-extend(LABEL << 2)

   LABEL        = low16((0x00400014 - (0x00400000 + 4)) >> 2)
                = low16(0x10 >> 2)             (16/ 4)
                = 0x4                      (forward 4 instructions)
```

Example of Addresses in Branches

- Branch address is "PC relative"
  - The target is RELATIVE to address of the branch instruction itself
  - Next PC = PC + 4 + sign-extend(16-bit immediate << 2)

```
0x00400000      LABEL:  andi   $s0,$s0,0xFF
0x00400004              srl    $s0,$s0,2
0x00400008              lui    $s1,0xFF00
0x0040000c              ori    $s1,$s0,$s1
0x00400010              beq    $s0,$s1,LABEL  # what's LABEL???
0x00400014              …
   PC when branch taken=0x00400010 + 4 + sign-extend(LABEL << 2)

   LABEL        = low16((0x00400000 - (0x00400010 + 4))) >> 2)
                = low16((0xFFFFFFEC) >> 2)     (-20 / 4)
                = 0xFFFB                   (backward -5 instructions)
```

J-format

Jump	op	26-bit immediate

- The address of next instruction is obtained from PC and the immediate value
  - Next address = {PC[31:28],IMM[25:0],00}
  - Address boundaries of 256MB

- Example
  - j 10000

2	2500

Control

- What about comparisons against constant?
  - Can we use “beqi”??? Why not?

- Comparison variants with an immediate
  - slti $s0,$s1,16
  - addi $s0,$s1,20 ($s0==0 when $s1==20)

```
if      (a > 20) { a = a + 2; }

        slti    $at,$s0,21      # $at==1 when a<=20
        bne     $at,$0,LAB      # $at!=0 implies $at==1, skip
        addi    $s0,$s0,2
LAB:    …                       # target when a<=20
```

Examples of control flow

- Weather programs
  - weather.c
  - weather.asm - illustrates if-then-else and while loop
  - weather2.asm - a slightly improved version (still if-then-else)
  - weather3.asm - illustrates a “computed goto” (switch)
  - weather4.asm - illustrates an algorithm change, using a table
- sam12.asm
  - convert a 32-bit number into hex characters, which are displayed with the OS print string service
- We’ll see many more examples!