

## CS306: Introduction to Perl

## Section 4: Control Structures

### Section #4: Control Structures

U. of Alabama at Birmingham  
Dept. of Computer & Information Sciences

Conditional structures  
Looping structures

Slide 1

Slide 2

### Conditional Structures - *if..elsif..else*

- `if (some condition) {`  
    ...  
    `} elsif (some other condition) {`  
    ...  
    `} else {`  
    ...  
    `}`

Slide 3

### Conditional Structures – *unless*

- `unless ($timeleft == 0) {`  
    `print "Keep playing.\n";`  
    `}`

Slide 4

## Conditional Structures - `? :` operator

- Perl's only trinary operator
  - condition ? then : else
  - `$string = $senior ? "congrats" : "see you next year";`

Slide 5

## Looping Structures – *while* and *until*

- `while (! $finished) {`
  - ...
  - `}`
- `until ($finished) {`
  - ...
  - `}`

Slide 6

## Looping Structures - *for*

- `for ($i = 0; $i < 10; $i++) {`
  - `# this code will run 10 times`
  - ...
  - `}`
- In Perl, this is really just a while loop
  - `$i = 0;`
    - `while ($i < 10) {`
      - ...
      - `$i++;`
      - `}`

Slide 7

## Looping Structures - *foreach*

- Best way to iterate over an array
- `foreach $day (@daysofweek) {`
  - `print "My favorite day is $day.\n";`
  - `}`
- `foreach $num (2..10) {`
  - `print "I have $num apples.\n";`
  - `}`
- `for $day (@daysofweek) { # Perl shortcut`

Slide 8

## *next and last*

- Use next and last to break out of loops
- while (\$i < 10) {  
    ...  
    if (\$done\_early) { last;}  
}
- foreach \$line (@lines) {  
    if (\$line eq "don't print me please") { next; }  
    print "\$line\n";  
}