

## Reading data from a file

- Problem: when to stop the loop?
- First entry in file could be count of number of items
  - problems: maintenance, large files
- Could use sentinal value
  - problem: may not be one, maintenance
- Want to automatically detect end of file

## Using >> to detect end of file

3

stream extraction operation produces a value:

int number; ifstream inputFile; inputFile.open("numbers.txt");

bool foundValue = (inputFile >> number);

#### • inputFile >> number:

- tries to read a value into number
- if it was successful, value is true
- if it failed (nothing left to input), value is false (and the value in number does not change)

## Using the result of >>

### • Example:

• • •

```
int number;
ifstream inputFile;
inputFile.open("numbers.txt");
bool foundValue = (inputFile >> number);
if (foundValue)
    cout << "The data read in was: " << number << endl;
else
    cout << "Could not read data from file." << endl;</pre>
```

### Can also use directly as relational expression:

if (inputFile >> number)

# Sum all the values in the file

```
int number;
ifstream inputFile;
inputFile.open("numbers.txt");
int total = 0;
```

```
while (inputFile >> number) {
   total = total + number;
}
cout << "The sum of the numbers in the file: " << total
        << endl;</pre>
```

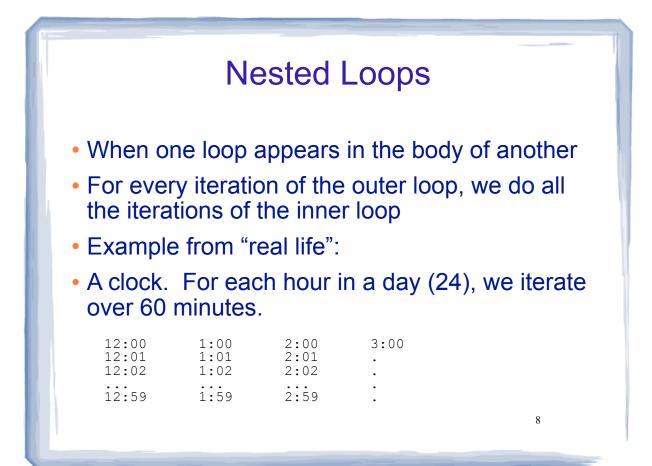
#### • Output:

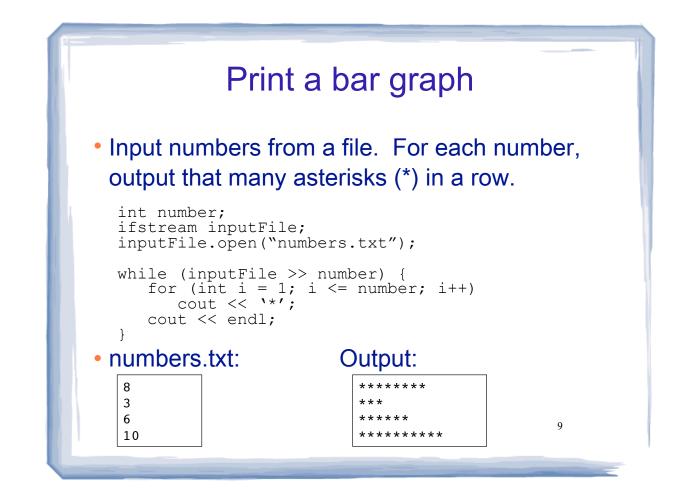
The sum of the numbers in the file: 344

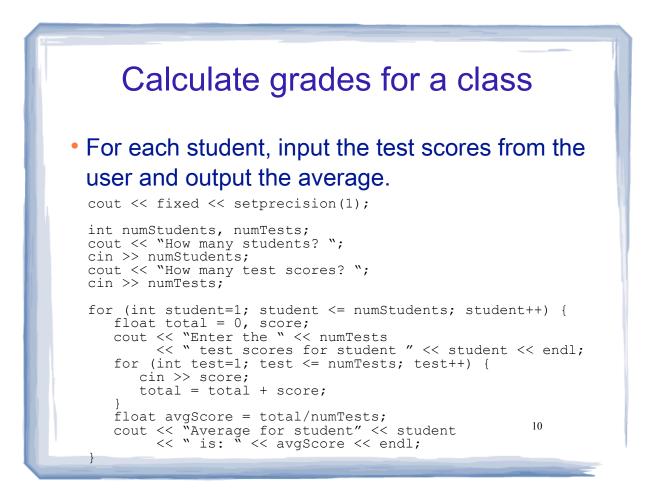
5

## Loops in C++: a summary

- Any loop can be made to work for a given problem
- while loop:
  - test at start of loop
  - generic
- for loop:
  - initialize/test/update
  - count-controlled loops
- do-while loop
  - always do at least once
  - good for repeating, simple menu processing







## Calculate grades for a class

• Output:

How many students? 3 How many test scores? 4 Enter the 4 test scores for student 1 88 90.5 92 77.5 Average for student1 is: 87.0 Enter the 4 test scores for student 2 66.5 70.5 80 86 Average for student2 is: 75.8 Enter the 4 test scores for student 3 99 93.5 80 79 Average for student3 is: 87.9

11

