Arrays

Chapter 8

page 471

Arrays (8.1)

- One variable that can store a *group of values of the same type*
    - Each value is called an *element* of the array
- Storing a number of related values
    - all grades for one student
    - all temperatures for one month
    - hours worked for each day

Arrays

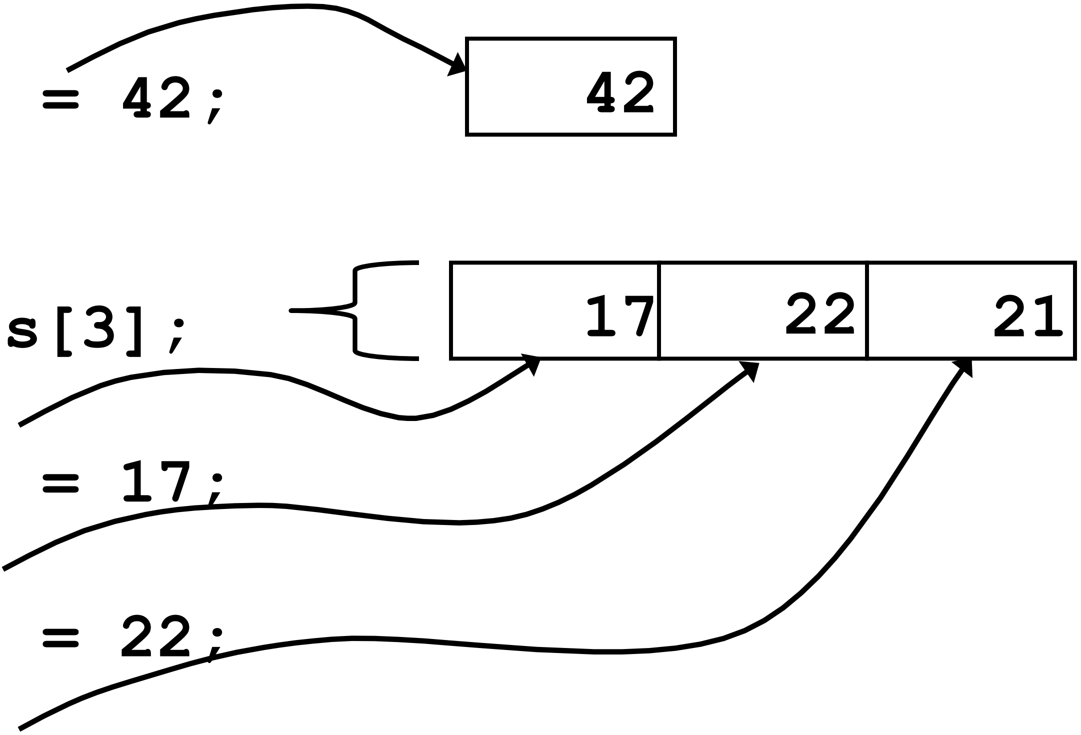
```
int age = 42;

int ages[3];

ages[0] = 17;

ages[1] = 22;

ages[2] = 21;
```



Arrays

```
int ages[3];
// datatype variable_name[size];

const int CLASSSIZE = 24;
string names[CLASSSIZE];
```

The size of the array must be a *literal* or a `const int`.

Using arrays (8.2)

- The first element in the array is the **0th** element!
- The ***index*** is an `int`

```
int y, x = 3;
int years[10];

years[0] = 2;
years[x] = 4;
y = years[0] + 9;
```

Practice

- Declare an array to hold the height, in inches, of six trees.
- Set the height of the trees as:
  - 32 inches
  - 45 inches
  - 99 inches
  - 120 inches
  - 500 inches
  - 600 inches

Practice (8.3)

- Write a snippet of code to read in 4 numbers from the user and put them in this array:

```
const int ARRAYSIZE = 4;
int vals[ARRAYSIZE];
```

- Print to the screen every value in the array
- Print the sum and average

Practice

- Read 20 exam scores from a file and print them in reverse order
- Ask the user for an exam number (0-19) and print that exam score to the screen
- Ask the user for an exam number and add 2 bonus points to that exam score.
- Find the max score in the array

Out of bounds (p 479)

- C++ does *not* check to make sure the ***index*** falls within the array
  - no *bounds checking*
  - this will cause unpredictable results!

Initialization (8.4)

- What is the equivalent of:

```
int value = 2; // initialize the variable

int tests[2] =

string names[3] =
```

- Initialize just a few values:

```
int value[4] =
```

Implicit array sizing (p 486)

- Set the size of the array by initializing it
- You *must* either specify a size or initialize the array

```
string names[] =


char letters[] =
```

Practice – parallel arrays

- Write a program to read the file below into *two arrays*. There are at most 100 students listed.

- Print the PUNetIDs of students who have a score between *88* and *100* (values supplied by the user).

Grades.txt

```
PUNetID  FinalAverage
AAAA1234 90.2
will4614 85.4
```

Practice - Continued

- Add 10 bonus points to **AAAA1234**
- Print out the overall class average