

Part I. Review Questions @ Page 589 – 591.

Qn. 1, 3, 4, 6, 9, 10, 12 – 14, 21, 23, 24, 27.

1. The *size declarator* indicates the number of elements, or values, an array can hold.
3. Each element of an array is accessed is accessed and indexed by a number known as a(n) *subscript*.
4. Subscript numbers in C++ always start at *0*.
6. C++ has no array *bounds* checking, which means you can inadvertently store data past the end of an array.
9. If the size declarator of an array definition is omitted, C++ counts the number of items in the *initialization list* to determine how large the array should be.
10. Look at the following array definition:
`double amount[5];`
 - A. How many elements does this array hold? *5*
 - B. What can you store in amount[5]? *Five double values.*
12. You cannot use the *assignment* operator to copy data from one array to another in a single statement.
13. Arrays are never passed to functions by *value* because there would be too much overhead in copying all the elements.
14. To pass an array to a function, pass the *name* of the array.
21. Look at the following array definition.
`int values[10];`
 - A. How many elements does this array have? *10*
 - B. What is the subscript of the first element of the array? *0*
 - C. What is the subscript of the last element of the array? *9*
 - D. If an int uses four bytes of memory, how much memory does the array use?
40 bytes.
23. Look at the following array definition.
`int numbers[5] = { 1, 2, 3 };`
 - A. What value is stored in numbers[2]? *3.*
 - B. What value is stored in numbers[4]? *0 if it's a global declaration, garbage if local.*

24. Assume that array1 and array2 are both 25-element integer arrays. Indicate whether each of the following statements is legal or illegal.
- array1 = array2; *Illegal.*
 - cout << array1; *Legal but prints the address of the first element of the array.*
 - cin >> array2; *Illegal.*
27. Look at the following array definition.
- ```
double sales[8][10];
```
- How many rows does the array have? *8 rows.*
  - How many columns does the array have? *10 columns.*
  - How many elements does the array have? *80 elements.*
  - Write a statement that stores 3.52 in the last column of the last row in the array. *sales[7][9] = 3.52;*

## Part II. Programming Challenge @ Page 595 - 596.

Qn. 10 – Baseball Champions.

Screenshot of Runtime:

### (1) Detroit Tigers

```
Hi!
This program looks up the number of times
a baseball team has won the World Series.

Here's a list of baseball teams.
Anaheim Angels
Arizona Diamondbacks
Atlanta Braves
Baltimore Orioles
Boston Americans
Boston Braves
Boston Red Sox
Brooklyn Dodgers
Chicago Cubs
Chicago White Sox
Cincinnati Reds
Cleveland Indians
Detroit Tigers
Florida Marlins
Kansas City Royals
Los Angeles Dodgers
Milwaukee Braves
Minnesota Twins
New York Giants
New York Mets
New York Yankees
Oakland Athletics
Philadelphia Athletics
Philadelphia Phillies
Pittsburgh Pirates
San Francisco Giants
St. Louis Cardinals
Toronto Blue Jays
Washington Senators

Please enter a team name: Detroit Tigers

The team, Detroit Tigers, has won
the World Series 2 time(s) from 1950 to 2014.

Process finished with exit code 0
```

### (2) Anaheim Angels

```
Hi!
This program looks up the number of times
a baseball team has won the World Series.

Here's a list of baseball teams.
Anaheim Angels
Arizona Diamondbacks
Atlanta Braves
Baltimore Orioles
Boston Americans
Boston Braves
Boston Red Sox
Brooklyn Dodgers
Chicago Cubs
Chicago White Sox
Cincinnati Reds
Cleveland Indians
Detroit Tigers
Florida Marlins
Kansas City Royals
Los Angeles Dodgers
Milwaukee Braves
Minnesota Twins
New York Giants
New York Mets
New York Yankees
Oakland Athletics
Philadelphia Athletics
Philadelphia Phillies
Pittsburgh Pirates
San Francisco Giants
St. Louis Cardinals
Toronto Blue Jays
Washington Senators

Please enter a team name: Anaheim Angels
Not found. Try again: Anaheim Angels

The team, Anaheim Angels, has won
the World Series 1 time(s) from 1950 to 2014.

Process finished with exit code 0
```

### (3) New York Yankees

```
Hi!
This program looks up the number of times
a baseball team has won the World Series.

Here's a list of baseball teams.
Anaheim Angels
Arizona Diamondbacks
Atlanta Braves
Baltimore Orioles
Boston Americans
Boston Braves
Boston Red Sox
Brooklyn Dodgers
Chicago Cubs
Chicago White Sox
Cincinnati Reds
Cleveland Indians
Detroit Tigers
Florida Marlins
Kansas City Royals
Los Angeles Dodgers
Milwaukee Braves
Minnesota Twins
New York Giants
New York Mets
New York Yankees
Oakland Athletics
Philadelphia Athletics
Philadelphia Phillies
Pittsburgh Pirates
San Francisco Giants
St. Louis Cardinals
Toronto Blue Jays
Washington Senators

Please enter a team name: New York Yankees

The team, New York Yankees, has won
the World Series 15 time(s) from 1950 to 2014.

Process finished with exit code 0
```

Source Code:

1. *BaseballChampions.h*
2. *BaseballChampions.cpp*
3. *main.cpp*

The source code is also stored at Github.

Link below:

[https://github.com/TheLoneWoof1102/FA17\\_CSC121001/tree/master/Source%20Code/Homework-Ch8.Qn10](https://github.com/TheLoneWoof1102/FA17_CSC121001/tree/master/Source%20Code/Homework-Ch8.Qn10)

### **main.cpp**

```
#include <iostream>
#include <string>
#include "BaseballChampions.h"
using namespace std;

int main() {
 BaseballChampions bs;
 string teamName;

 if (!bs.good()) {
 cout << "There's something wrong with the files.";
 return -1;
 }

 cout << "Hi!" << endl;
 cout << "This program looks up the number of times " << endl
 << "a baseball team has won the World Series." << endl;
 cout << endl;

 cout << "Here's a list of baseball teams. " << endl;
 for (string team : bs.getTeams()) {
 cout << " " << team << endl;
 }
 cout << endl;

 cout << "Please enter a team name: ";
 getline(cin, teamName);
 while (!bs.validateTeamName(teamName)) {
 cout << " Not found. Try again: ";
 getline(cin, teamName);
 }
 cout << endl;

 cout << "The team, " << teamName << ", has won " << endl
 << "the World Series " << bs.countTeamWins(teamName)
 << " time(s) from 1950 to 2014." << endl;
 cout << endl;

 return 0;
}
```

**// END of main.cpp.**

## BaseballChampions.h

```
//
// Created by TheLoneWoof on 10/9/17.
//

#ifndef HOMEWORK_CH8_QN10_BASEBALLCHAMPIONS_H
#define HOMEWORK_CH8_QN10_BASEBALLCHAMPIONS_H

#include <string>
#include <vector>

class BaseballChampions {

private:
 bool isGood = false;
 std::vector<std::string> teams;
 std::vector<std::string> raw_championHistory;
 bool init();

public:
 BaseballChampions();
 bool good();
 std::vector<std::string> getTeams();

 bool validateTeamName(std::string);
 int countTeamWins(std::string);
};

#endif //HOMEWORK_CH8_QN10_BASEBALLCHAMPIONS_H
```

**// END of *BaseballChampions.h*.**

## BaseballChampions.cpp

```
//
// Created by TheLoneWoof on 10/9/17.
//

#include "BaseballChampions.h"
#include <iostream>
#include <fstream>
using namespace std;

bool BaseballChampions::init() {
 ifstream teamFile, listFile;
 string tempStr;
 teamFile.open("teams.txt");

 if (teamFile) {
 while (teamFile.good()) {
 getline(teamFile, tempStr);
 // Removes '\r' characters from Mac text files.
 // string::pop_back() removes last character.
 if (tempStr.back() == '\r') {
 tempStr.pop_back();
 }
 teams.push_back(tempStr);
 }
 teamFile.close();
 } else {
 return false; // Fail.
 }

 listFile.open("WorldSeriesWinners.txt");
 if (listFile) {
 while (listFile.good()) {
 getline(listFile, tempStr);
 // Removes '\r' characters from Mac text files.
 // string::pop_back() removes last character.
 if (tempStr.back() == '\r') {
 tempStr.pop_back();
 }
 raw_championHistory.push_back(tempStr);
 }
 listFile.close();
 } else {
 return false; // Fail
 }

 return true;
}

BaseballChampions::BaseballChampions() {
 isGood = init();
}

bool BaseballChampions::good() {
 return isGood;
}
```

*// BaseballChampions.cpp is continued next page.*

### BaseballChampions.cpp – cont'd.

```
vector<string> BaseballChampions::getTeams() {
 return teams;
}

bool BaseballChampions::validateTeamName(string tryTeam) {
 bool isFound = false;
 for (string team : teams) {
 if (team == tryTeam) {
 isFound = true;
 break;
 }
 }
 return isFound;
}

int BaseballChampions::countTeamWins(string teamName) {
 int winCount = 0;
 for (string teamWin : raw_championHistory) {
 if (teamWin == teamName) {
 winCount++;
 }
 }
 return winCount;
}
```

**// END OF *BaseballChampions.cpp*.**