

Chapter 9 Arrays and Strings

1. Be able to declare, initialize, and manipulate individual elements of a one-dimensional array.
2. Be able to pass one-dimensional arrays to functions.
3. Be able to use multiple-dimensional arrays.
4. Be able to declare, initialize, and manipulate individual elements of a two-dimensional array.
5. Be able to pass two-dimensional arrays to functions.
6. Be able to pass one row of a two-dimensional array to a function.

Recommended exercises pp. 485-490 #1,2,5,9,13,14,15

Recommended programming exercises pp. 490-491 #1,5,12

Chapter 10 Application of Arrays and the vector Type

1. Be able to use at least one (preferably at least two) sorting technique(s) to rearrange data in an array.
2. Be able to search an array using both linear and binary searching techniques.

Note: you will be given uncommented copies of the search and sort functions (same as with the quizzes).

Recommended exercises pp. 547-549 #1 – 6

Recommended programming exercises pp. 549-551 #1,2,3,5,7

Chapter 11 Records (Structs)

1. Be able to declare a struct and manipulate its fields.
2. Be able to declare an array of structs and manipulate the elements.

Recommended exercises pp. 589-591 #1-4

Recommended programming exercises pp. 573-574 #1

Chapter 12 Classes and Data Abstraction

1. Understand the differences between structs and classes.
2. Be able to create and use class objects.
3. Be able to declare a class and control access to class data members.
4. Be able to create and use constructor and destructor functions.
5. Be able to design and code a program which includes a user-created class.

Recommended exercises pp. 664-669 #1-9

Recommended programming exercise pp. 669-671 #1,2

Chapter 13 Inheritance and Composition

1. Be able to declare a class which inherits from another class using public inheritance.
2. Understand what kind of access the derived class has to public, protected, and private members of the base class.
3. Understand what kind of access a client of the derived class has to public, protected, and private members of the derived class and of the base class.
4. Understand how the constructor of the derived class can call the constructor of the base class.
5. Be able to declare a class (x, for example) which includes a variable of another class type (y, for example) – this is called composition.
6. Understand what kind of access the class x functions have to variables and functions in class y.
7. Understand what kind of access a client of class x has to variables and functions in class y and

class x.

Recommended exercises pp. 731-737 #1, 3-11

Recommended programming exercises pp. 737-740 #1,3,4,5

Chapter 14 Pointers, Classes and Virtual Functions

1. Be more familiar with the pointer data type and pointer variables.
2. Be able to declare and manipulate pointer variables.
3. Be able to use the address of operator and dereferencing.
4. Know what dynamic variables and how to create and destroy memory for them using **new** and **delete**.
5. Know what virtual functions are and how they are useful in declaring abstract base classes for inheritance hierarchies.

Recommended exercises pp. 794-800 #1,2,11,17,18

Recommended programming exercises pp. 800 #1

Chapter 15 Overloading and Templates

1. Be able to create and use friend functions.
2. Be able to overload functions and call them.
3. Be able to overload operator functions and call them.
4. Understand the differences between a member function and a friend function.
5. Understand what kind of access a friend function has to private variables in the class.
6. Understand the restrictions on overloading operator functions.

Recommended exercises pp. 879-883 #1 a-g,i #2,4,6,8,9

Recommended programming exercises #1,10