Chapter 8

Algorithms

OBJECTIVES

- Understand the concept of an algorithm.
- Define and use the three constructs for developing algorithms: sequence, decision, and repetition.
- Understand and use three tools to represent algorithms: flowchart, pseudocode, and structure chart.
- Understand the concept of modularity and subalgorithms.
- List and comprehend common algorithms.

8.1 CONCEPT

Informal Definition

- Algorithm: a step-by-step method for solving a problem or doing a task

Figure 8-1

Informal definition of an algorithm used in a computer

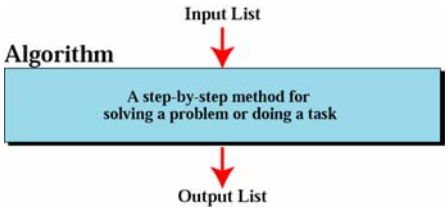


Figure 8-2

Finding the largest integer among five integers

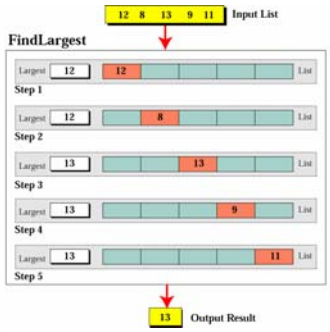


Figure 8-3

Defining actions in FindLargest algorithm

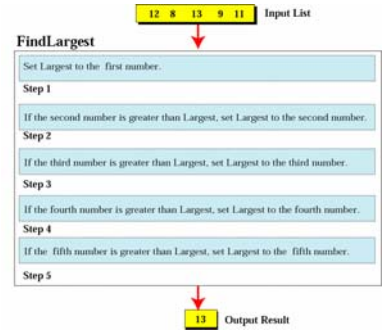


Figure 8-4

FindLargest refined

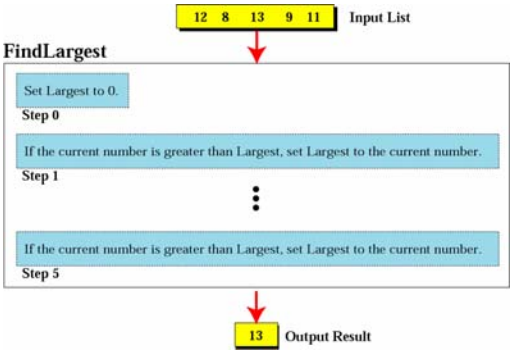
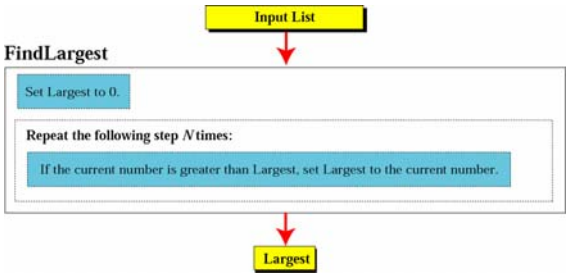


Figure 8-5

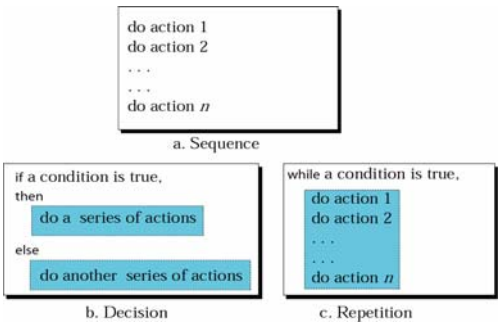
Generalization of FindLargest



8.2 THREE CONSTRUCTS

Figure 8-6

Three constructs



8.3 ALGORITHM REPRESENTATION

Figure 8-7

Flowcharts for three constructs

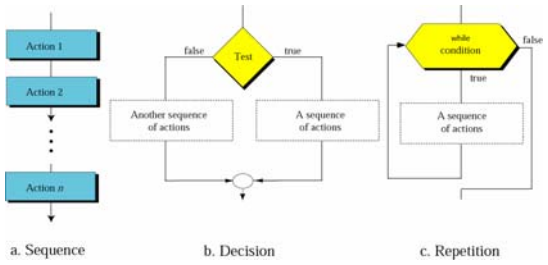
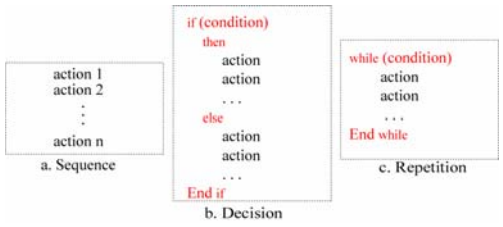


Figure 8-8

Pseudocode for three constructs



Example 1

Write an algorithm in pseudocode that finds the average of two numbers

Solution

See Algorithm 8.1 on the next slide.

Algorithm 8.1: Average of two

```
AverageOfTwo
Input: Two numbers
1. Add the two numbers
2. Divide the result by 2
3. Return the result by step 2
End
```


Example 2

Write an algorithm to change a numeric grade to a pass/no pass grade.

Solution

See Algorithm 8.2 on the next slide.

Algorithm 8.2: Pass/no pass Grade

```
Pass/NoPassGrade
Input: One number
1. if (the number is greater than or equal to 70)
   then
      1.1 Set the grade to "pass"
   else
      1.2 Set the grade to "nopass"
   End if
2. Return the grade
End
```

Example 3

Write an algorithm to change a numeric grade to a letter grade.

Solution

See Algorithm 8.3 on the next slide.

Algorithm 8.3: Letter grade

```
LetterGrade
Input: One number
1. if (the number is between 90 and 100, inclusive)
   then
      1.1 Set the grade to "A"
   End if
2. if (the number is between 80 and 89, inclusive)
   then
      2.1 Set the grade to "B"
   End if
```

Continues on the next slide

Algorithm 8.3: Letter grade (continued)

```
3. if (the number is between 70 and 79, inclusive)
   then
      3.1 Set the grade to "C"
   End if
4. if (the number is between 60 and 69, inclusive)
   then
      4.1 Set the grade to "D"
   End if
```

Continues on the next slide

Algorithm 8.3: Letter grade (continued)

```
5. If (the number is less than 60)
   then
      5.1 Set the grade to "F"
   End if
6. Return the grade
   End
```


Example 4

Write an algorithm to find the largest of a set of numbers. You do not know the number of numbers.

Solution

See Algorithm 8.4 on the next slide.

Algorithm 8.4: Find largest

```
FindLargest
Input: A list of positive integers
1. Set Largest to 0
2. while (more integers)
     2.1 if (the integer is greater than Largest)
           then
             2.1.1 Set largest to the value of the integer
           End if
   End while
3. Return Largest
   End
```

Example 5

Write an algorithm to find the largest of 1000 numbers.

Solution

See Algorithm 8.5 on the next slide.

Algorithm 8.5: ***Find largest of 1000 numbers***

```
FindLargest
Input: 1000 positive integers
1. Set Largest to 0
2. Set Counter to 0
3. while (Counter less than 1000)
     3.1 if (the integer is greater than Largest)
           then
             3.1.1 Set Largest to the value of the integer
         End if
     3.2 Increment Counter
   End while
4. Return Largest
   End
```


Summary

- An algorithm is a step-by-step method for solving a problem or doing a task
- An algorithm accepts an input list of data and creates an output list of data
- A program is a combination of sequence constructs, decision constructs, and repetition constructs
- A flowchart is a pictorial representation of an algorithm
- Pseudocode is an Englishlike representation of an algorithm