Metodi vodjeni tabelama

Sadrzaj

- Sta su metode vodjene tabelama?
- Opsta razmatranja u koriscenju metoda vodjenih tabelama
- Direktan pristup tabelama
- Pristup tabelama pomocu indeksa
- Stepenast pristup tabelama(Stair-Step)

Opsta razmatranja u koriscenju metoda vodjenih tabelama

Java Example of Using Complicated Logic to Classify a Character

```
if ( ( ( 'a' <= inputChar ) && ( inputChar <= 'z' ) ) ||
   ( ( 'A' <= inputChar ) && ( inputChar <= 'Z' ) ) ) {
   charType = CharacterType.Letter;
}
else if ( ( inputChar == ' ' ) || ( inputChar == ',' ) ||
   ( inputChar == '.' ) || ( inputChar == '!' ) || ( inputChar == '(' ) ||
   ( inputChar == ')' ) || ( inputChar == ':' ) || ( inputChar == ';' ) ||
   ( inputChar == '?' ) || ( inputChar == '-' ) ) {
   charType = CharacterType.Punctuation;
}
else if ( ( '0' <= inputChar ) && ( inputChar <= '9' ) ) {
   charType = CharacterType.Digit;
}
```

Java Example of Using a Lookup Table to Classify a Character

```
charType = charTypeTable[ inputChar ];
```

- Dva pitanja u vezi koriscenja metoda vodjenih tabelama

Direktan pristup tabelama

- Primer: Dani u mesecu

Visual Basic Example of a Clumsy Way to Determine the Number of Days in a Month

```
If ( month = 1 ) Then
   days = 31
ElseIf ( month = 2 ) Then
   days = 28
ElseIf ( month = 3 ) Then
   days = 31
ElseIf ( month = 4 ) Then
   days = 30
ElseIf ( month = 5 ) Then
   days = 31
ElseIf ( month = 6 ) Then
   days = 30
ElseIf ( month = 7 ) Then
   days = 31
ElseIf ( month = 8 ) Then
   days = 31
ElseIf ( month = 9 ) Then
   days = 30
ElseIf ( month = 10 ) Then
   days = 31
ElseIf ( month = 11 ) Then
   days = 30
ElseIf ( month = 12 ) Then
   days = 31
End If
```

Visual Basic Example of an Elegant Way to Determine the Number of Days in a Month

```
' Initialize Table of "Days Per Month" Data
Dim daysPerMonth() As Integer = _
   { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 }
```

Visual Basic Example of an Elegant Way to Determine the Number of Days in a Month (continued)

```
days = daysPerMonth( month-1 )
```

Visual Basic Example of an Elegant Way to Determine the Number of Days in a Month (continued)

```
days = daysPerMonth( month-1, LeapYearIndex() )
```

Direktan pristup tabelama

- Primer 2: Odredjivanje stopa medicinskog osiguranja

Java Example of a Clumsy Way to Determine an Insurance Rate

```
if ( gender == Gender.Female ) {
   if ( maritalStatus == MaritalStatus.Single ) {
      if ( smokingStatus == SmokingStatus.NonSmoking ) {
         if ( age < 18 ) {
            rate = 200.00;
         }
         else if ( age == 18 ) {
            rate = 250.00;
         }
         else if ( age == 19 ) {
            rate = 300.00;
         }
         ...
         else if ( 65 < age ) {
            rate = 450.00;
      }
      else {
         if ( age < 18 ) {
            rate = 250.00;
         }
        else if ( age == 18 ) {
           rate = 300.00;
        }
        else if ( age == 19 ) {
           rate = 350.00;
        ...
        else if ( 65 < age ) {
           rate = 575.00;
        }
     }
   else if ( maritalStatus == MaritalStatus.Married )
   ...
}
```

Visual Basic Example of Declaring Data to Set Up an Insurance-Rates Table

```
Public Enum SmokingStatus
   SmokingStatus_First = 0
   SmokingStatus_Smoking = 0
   SmokingStatus_NonSmoking = 1
   SmokingStatus_Last = 1
End Enum

Public Enum Gender
   Gender_First = 0
   Gender_Male = 0
   Gender_Female = 1
   Gender_Last = 1
End Enum

Public Enum MaritalStatus
   MaritalStatus_First = 0
   MaritalStatus_Single = 0
   MaritalStatus_Married = 1
   MaritalStatus_Last = 1
End Enum

Const MAX_AGE As Integer = 125
```

Visual Basic Example of an Elegant Way to Determine an Insurance Rate

```
Dim rateTable ( SmokingStatus_Last, Gender_Last, MaritalStatus_Last, _
   MAX_AGE ) As Double
```

Once you declare the array, you have to figure out some way of putting data into it. You can use assignment statements, read the data from a disk file, compute the data, or do whatever is appropriate. After you've set up the data, you've got it made when you need to calculate a rate. The complicated logic shown earlier is replaced with a simple statement like this one:

Visual Basic Example of an Elegant Way to Determine an Insurance Rate

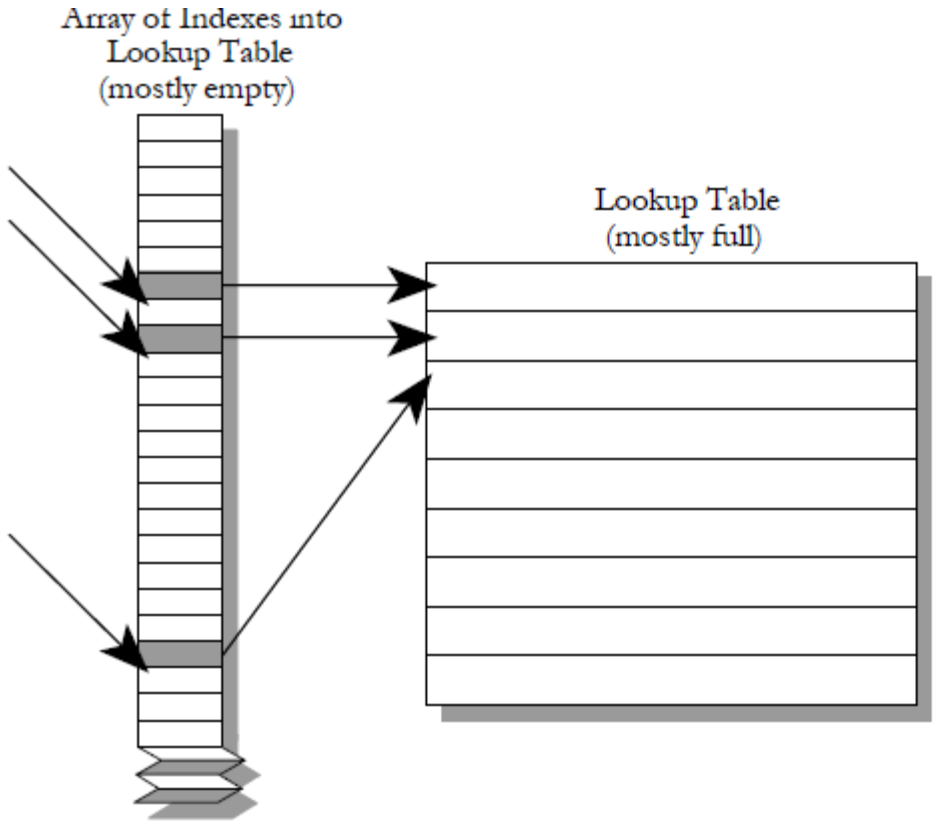
```
rate = rateTable( smokingStatus, gender, maritalStatus, age )
```

Direktan pristup tabelama

- Petljanje sa kljucevima za pretrazivanje
  - dupliranje informacija da bi kljuc radio direktno
  - transformacija kljuca da bi radio direktno
  - izolovani kljuc – transformacija u sopstvenu funkciju

Pristup tabelama pomocu indeksa

- Sta je ?



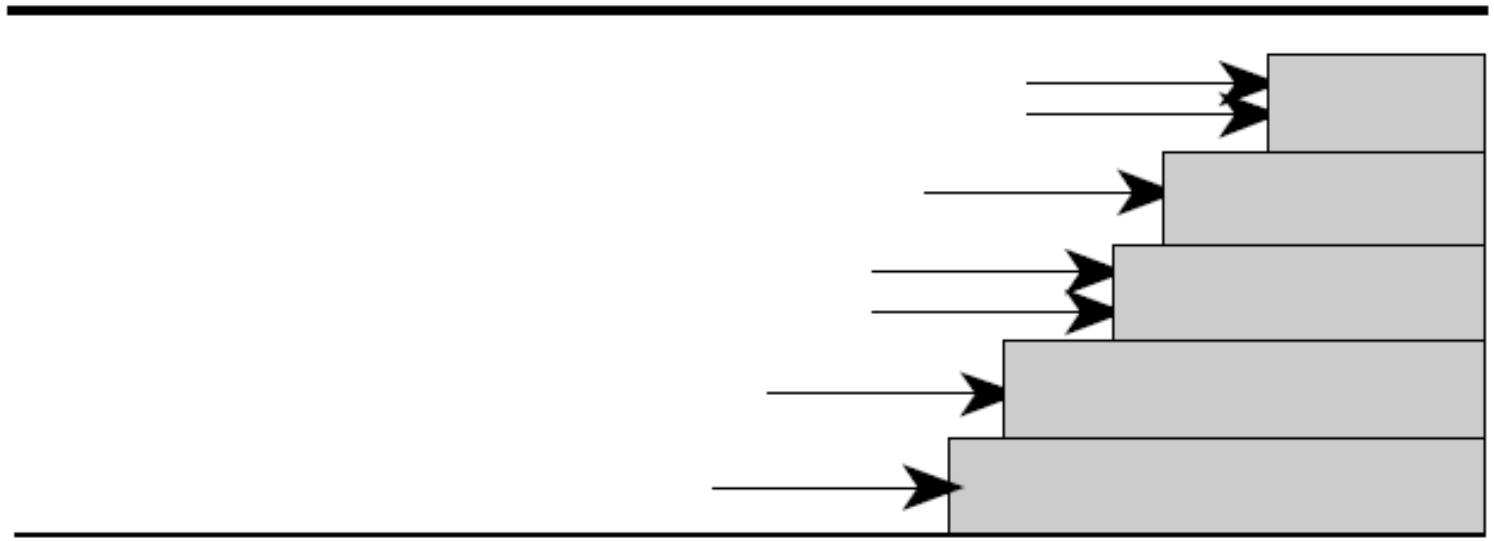
F18xx04

Figure 18-4

Rather than being accessed directly, an indexed access table is accessed via an intermediate index.

Stepenast pristup tabelama (Stair-Step)

- Sta je?



F18xx05

Figure 18-5

The stair-step approach categorizes each entry by determining the level at which it hits a "staircase." The "step" it hits determines its category.

Visual Basic Example of a Stair-Step Table Lookup

```
' set up data for grading table
Dim rangeLimit() As Double = { 50.0, 65.0, 75.0, 90.0, 100.0 }
Dim grade() As String =      { "F",  "D",  "C",  "B",   "A"  }
maxGradeLevel = grade.Length - 1
...

' assign a grade to a student based on the student's score
gradeLevel = 0
studentGrade = "A"
While ( ( studentGrade = "A" ) and ( gradeLevel < maxGradeLevel ) )
   If ( studentScore < rangeLimit( gradeLevel ) ) Then
      studentGrade = grade( gradeLevel )
   End If
   gradeLevel = gradeLevel + 1
Wend
```

Kraj

- *Hvala na paznji !!!*