

C8-1

Algorithm

1. Use a subtype to represent the numbers for months
2. Use an enumeration to represent the named months
3. Use an enumeration to represent the roman months
4. Get the inputs from the user
5. Convert the month into roman and named formats using
 - a. `New_Type_Package'Val(Month_Type'Pos(Month) -1);`
6. Display the months in all three formats to the user.

Note: The enumerations range from 0 to (number_of_elements_in_Enumeration -1).

Code Listing

GNAT 3.13p (20000509) Copyright 1992-2000 Free Software Foundation, Inc.

Compiling: c:/docume~2/jk/desktop/16070/codeso~1/translate_dates.adb (source file time stamp: 2003-09-24 01:35:00)

```
1. -----
2. -- Program to accept different date formats
3. -- Programmer : Jayakanth Srinivasan
4. -- Date Last Modified : 09-23-2003
5. -----
6.
7. with Ada.Text_IO;
8. with Ada.Integer_Text_IO;
9.
10. procedure Translate_Dates is
11.   --use a subtype to limit the date to be between 1 and 31.
12.   subtype Date_Type is Integer range 1..31;
13.   -- use a subtype to limit the month to be between 1 and 12
14.   subtype Month_Type is Integer range 1..12;
15.
16.   -- define an enumeration type for roman months
17.   type Roman_Month_Type is
18.     (I,
19.      Ii,
20.      Iii,
21.      Iv,
22.      V,
23.      Vi,
24.      Vii,
25.      Viii,
26.      Ix,
27.      X,
28.      Xi,
29.      Xii);
30.
31.   -- define an enumeration type for names of months
32.   type Named_Month_Type is
```

```

33.     (January,
34.     February,
35.     March,
36.     April,
37.     May,
38.     June,
39.     July,
40.     August,
41.     September,
42.     October,
43.     November,
44.     December);
45.
46. -- create a package that can print the month in roman numerals
47. package Roman_Month_Io is new Ada.Text_Io.Enumeration_Io(Enum => Roman_Month_Type);
48.
49. -- create a package that can print the month's name
50. package Named_Month_Io is new Ada.Text_Io.Enumeration_Io(Enum => Named_Month_Type);
51.
52. Year      : Integer;
53. Date      : Date_Type;
54. Month     : Month_Type;
55. Roman_Month : Roman_Month_Type;
56. Named_Month : Named_Month_Type;
57.
58. begin
59.   Ada.Text_Io.Put("Please Enter the Date 1..31 : ");
60.   Ada.Integer_Text_Io.Get(Date);
61.   Ada.Text_Io.New_Line;
62.
63.   Ada.Text_Io.Put("Please Enter the Month 1 ..12 : ");
64.   Ada.Integer_Text_Io.Get(Month);
65.   Ada.Text_Io.New_Line;
66.
67.   Ada.Text_Io.Put("Please Enter the Year: ");
68.   Ada.Integer_Text_Io.Get(Year);
69.   Ada.Text_Io.New_Line;
70.
71. -- convert the month into roman month
72. Roman_Month := Roman_Month_Type'Val(Month_Type'Pos(Month)-1);
73.
74. -- convert the month into the name
75. Named_Month := Named_Month_Type'Val(Month_Type'Pos(Month)-1);
76.
77. -- display the dates in regular format
78. Ada.Text_Io.Put(" i. ");
79. Ada.Integer_Text_Io.Put(Date);
80. Ada.Text_Io.Put(" / ");
81. Ada.Integer_Text_Io.Put(Month);
82. Ada.Text_Io.Put(" / ");
83. Ada.Integer_Text_Io.Put(Year);
84. Ada.Text_Io.New_Line;
85.
86. -- display the date in named format
87. Ada.Text_Io.Put(" ii. ");
88. Ada.Integer_Text_Io.Put(Date);
89. Ada.Text_Io.Put(" ");
90. Named_Month_Io.Put(Named_Month);

```

```

91. Ada.Text_Io.Put(" ");
92. Ada.Integer_Text_Io.Put(Year);
93. Ada.Text_Io.New_Line;
94.
95. -- display the date in roman format
96. Ada.Text_Io.Put(" iii. ");
97. Ada.Integer_Text_Io.Put(Date);
98. Ada.Text_Io.Put(".");
99. Roman_Month_Io.Put(Roman_Month);
100. Ada.Text_Io.Put(".");
101. Ada.Integer_Text_Io.Put(Year);
102. Ada.Text_Io.New_Line;
103.
104. end Translate_Dates;

```

104 lines: No errors

C-8 2. What are the First and Last values of the following data types

a. Integer

Integer'First = -2147483648
Integer'Last = 2147483647

b. Float

Float'First = -3.40282E+38
Float'Last = 3.40282E+38

c. Character

Character'First =
Character'Last =

Note that both the character values are control character and hence do not get printed on the screen. The position values are 0, 255

d. Boolean

Boolean'First = FALSE
Boolean'Last = TRUE