```
 1: //shows Swap Sorting & BinarySearch in Array _138
 2: //runs under Win and Linux (CLX), locs=68
 3: //*******************************************************
 4: //resourcestring atmega='TMyArray' and Sorting_Visual_Delphi.exe;
 5:
 6: program SwapSort_Bubble_BinSearch;
 7:
 8: const
 9:   ArSize = 20;
10:
11: type
12:   TMyArray = array[1..ArSize] of integer;
13:
14: var
15:   MyArray: TMyArray;
16:       i: integer;
17:
18:
19: Procedure SwapSort;
20: var
21:   temp,i,j: integer;
22: Begin
23:    { Compare every element in the array ... }
24:    for i:= 1 to ArSize do begin
25:       { ... to every other element in the array ... }
26:       for j:= i+1 to ArSize do begin
27:          if MyArray[i] > MyArray[j] then begin
28:             { ... and swap them if they're out of order. }
29:             temp:= MyArray[i];
30:             MyArray[i]:= MyArray[j];
31:             MyArray[j]:= temp;
32:          end;
33:       end; {j}
34:    end; {i}
35: End;
36:
37: Function binSearch(A: TmyArray; theWord: integer; left,right:
    integer): integer;
38: var
39:  mid: integer;
40: begin
41:   while (left <= right) do begin
42:     mid:= (left + right) div 2;
43:     if A[mid] = theWord then begin
44:       result:= A[mid];
45:       writeln('Found!');
46:       Break;
47:     end
48:     if A[mid] > theWord then right:= mid - 1;
49:     if A[mid] < theWord then left:= mid + 1;
50:   end;
51: end;
52:
53:
54: Begin  {Main}
```

```
55:    Randomize;   {Init random number function.}
56:    writeln('Before . . .');
57:    for i:= 1 to ArSize do begin
58:       MyArray[i]:= trunc(Random(10000))+1;
59:       writeln(inttoStr(i)+ ': '+IntToStr(MyArray[i]));
60:    end;
61:    SwapSort;
62:    writeln('After . . .');
63:    for i:= 1 to ArSize do
64:       writeln(inttoStr(i)+ ': '+IntToStr(MyArray[i]));
65:    Writeln('BinSearch Found.. '+
66:              IntToStr(binSearch(myarray,myarray[18],1,ArSize)));
67:    SearchAndOpenDoc(ExePath+'exercices\sorting_visual_delphi.exe')

68: End.  {App}
69:
70:
71: ------------------------------------------------------------------
    -----
72: The fourth step is to "sort the list".  What does that mean?  Well,
    later in the same article I presented a more refined outline of the
    program which included a procedure entitled "SortList".  In it we
    asked the user about how they wanted the data sorted and then
    called PleaseWait.  The next line of code read "
    list }".  Not much help there.  So this is the topic for today:
    Sorting.
73:
74: You can sort anything that's in a list.  There are many ways to
    represent a list: an array, a chain of pointers, an instance of
    Delphi's TStringList object.  For the purposes of this discussion
    and demonstration, we
    dimensional array.  There are several well known and well
    established ways to sort data.  We'll look at two of them.  First,
    the easy-to-understand but slow way: the Swap Sort.  Second is the
    harder-to-understand but fast way: the QuickSort.
75:
76: //memo1 is script editor
77: //memo2 is output space
78: //max@kleiner.com
```