

```

1: //((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((
2: // How do it smarter with less code, solves the number running rule
3: //((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((
4:
5: program UMLRunner;
6:
7:
8: function teststring(vstring: string): boolean;
9: var b, i: integer;
10: begin
11:   for i:= 1 to length(vstring) do begin
12:     b:= b + strToInt(vstring[i]);
13:   end
14:   memo2.lines.add(intToStr(b))
15:   if b= 55 then result:= true;
16: end;
17:
18: procedure Display(var vshow, rword: string; cntr, i: integer);
19: begin
20:   Rword:= Rword + intToStr(cntr) + vshow[i]
21: end;
22:
23: function runString(Vshow: string): string;
24: var rword: string;
25:   cntr,i: integer;
26: begin
27:   cntr:=1;
28:   Rword:=''; //initialize
29:   for i:= 1 to length(vshow) do begin
30:     if i= length(vshow) then
31:       Display(vshow, rword, cntr, i)
32:     else if
33:       (vshow[i]=vshow[i+1]) then inc(cntr)
34:     else begin
35:       if cntr = 1 then cntr:= 1;
36:       Display(vshow, rword, cntr, i)
37:       //reinit counter!
38:       cntr:= 1;
39:     end;
40:   end; //loop
41:   result:= Rword;
42: end;
43:
44: const
45:   RUNSTR = '112222778888999911116';
46: begin
47:   memo2.font.size:= 14;
48:   writeln(runstr)
49:   memo2.lines.add(runString(RUNSTR))
50:   if teststring(runstring(RUNSTR)) then
51:     memo2.lines.add('OK') else
52:     memo2.lines.add('not OK')
53: end.

```