```
 1 with Ada.Float_Text_IO;         use Ada.Float_Text_IO;
 2 with Ada.Text_IO;               use Ada.Text_IO;
 3 with Ada.Integer_Text_IO;       use Ada.Integer_Text_IO;
 4
 5 with Ada.Numerics.Elementary_Functions;
 6 use Ada.Numerics.Elementary_Functions;
 7
 8 procedure reduction is
 9
10     type vector is array(integer range <>) of float;
11
12     length: constant integer := 1024;
13
14     a0, d0, x0: vector(1..length);
15
16     testa: vector(1..length);
17     testx: vector(1..length);
18     testd: vector(1..length);
19
20     x00: float := 5.6;
21
22     procedure fill is
23     begin
24         for i in 1..length loop
25             a0(i):=float(i)/5000.0;
26             d0(i):=float(2*i)/500.0;
27             testa := a0;
28             testd := d0;
29         end loop;
30     end fill;
31
32     procedure test is
33         t: integer := 0;
34         eps : constant float := 0.01;
35     begin
36         testx:=(1=>testa(1)*x00+testd(1), others=>0.0);
37         for i in 2..length loop
38             testx(i):=testa(i)*testx(i-1) + testd(i);
39         end loop;
40         for i in 1..length loop
41             if abs(x0(i)-testx(i)) > eps then
42                 put("Answers don't match:");
43                 put(i);
44                 put(x0(i), 8,5);
45                 put(testx(i),8,5);
46                 new_line;
47                 t := t + 1;
48             end if;
49         end loop;
50         if t = 0 then
51             put("Ok");
52         end if;
53     end test;
54
55     procedure red(a: in out vector; d: in out vector; x: in out vector) is
56         v1: vector(1..length);
57         v2: vector(1..length);
58         operation: integer;
59         shift: integer;
60
61         task type item is
62             entry set(i: in integer);
```

```
63              entry calculate;
64              entry check;
65          end item;
66
67          unit: array(1..length) of item;
68
69          task body item is
70              id: integer;
71          begin
72              accept set(i: in integer) do
73                  id:=i;
74              end set;
75              v1(id) := 0.0;
76              v2(id) := 0.0;
77              loop
78                  select
79                      accept calculate;
80                      case operation is
81                      when 1 =>
82                          v2(id):=v1(id)+v2(id);
83                      when 2 =>
84                          v2(id):=v1(id)*v2(id);
85                      when 0 =>
86                          if id - shift>0 then
87                              v2(id):=v1(id-shift);
88                          else
89                              v2(id):=0.0;
90                          end if;
91                      when others =>
92                          null;
93                      end case;
94                  or
95                      accept check;
96                  or
97                      terminate;
98                  end select;
99              end loop;
100         end item;
101
102         function log2(n: in integer) return integer is
103             ans, pow2: integer;
104         begin
105             pow2 := 1;
106             ans := 0;
107             while pow2 < n loop
108                 pow2 := pow2*2;
109                 ans := ans+1;
110             end loop;
111             return ans;
112         end log2;
113
114         procedure init is
115         begin
116             for i in 1..length loop
117                 unit(i).set(i);
118             end loop;
119         end init;
120
121         procedure step is
122         begin
123             for i in 1..length loop
124                 unit(i).calculate;
```

```
125             end loop;
126             for i in 1..length loop
127                 unit(i).check;
128             end loop;
129         end step;
130
131         function "+"(vv1: in vector; vv2: in vector) return vector is
132         begin
133             v1 := vv1;
134             v2 := vv2;
135             operation := 1;
136             step;
137             return v2;
138         end "+";
139
140         function "*"(vv1: in vector; vv2: in vector) return vector is
141         begin
142             v1 := vv1;
143             v2 := vv2;
144             operation := 2;
145             step;
146             return v2;
147         end "*";
148
149         function shiftr(v: in vector; l: in integer) return vector is
150         begin
151             v1 := v;
152             shift := l;
153             operation := 0;
154             step;
155             return v2;
156         end shiftr;
157
158     begin
159         init;
160         for l in 1..log2(length) loop
161             x:= a*shiftr(x, 2**(l-1)) + x;
162             a:= a*shiftr(a, 2**(l-1));
163         end loop;
164     end red;
165
166 begin
167     fill;
168     x0 := d0;
169     x0(1) := a0(1)*x00 + d0(1);
170     red(a0, d0, x0);
171     test;
172 end reduction;
173
174 --Таракчян  Левон  K5-224
175 --Вывод  программы  :
176 --Ok
177
```