

```

> # "
> c := proc(n :: integer)
  description "List a Collatz sequence (or hailstone trajectory) for a given integer input";
  local t;
  t := n;
  while t > 1 do
    print(t);
    if mod(t, 2) = 0 then t :=  $\frac{t}{2}$  else t := 3 * t + 1; end if;
  end do;
end proc;
c := proc(n::integer)
  local t;
  description "List a Collatz sequence (or hailstone trajectory) for a given integer input";
  t := n;
  while 1 < t do
    print(t); if mod(t, 2) = 0 then t := 1/2 * t else t := 3 * t + 1 end if
  end do
end proc

```

(1)

```

> one := c(6);
6
3
10
5
16
8
4
2
one := 1
> nextOne := c(22);
22
11
34
17
52
26
13
40
20
10
5
16
8
4

```

(2)

2

nextOne := 1

(3)

[> # notice branch at 10. It can come from 20 or 3.

[> # Matt C. Anderson

[> # 3-23-2017

[>