Computer Science and Mathematics
Summer term 2018

Solutions to selected exercises from Computer Science, part 2

Slide 86, task 4

The following Java method **f** gets an integer array **x** and the length **n** of the array as arguments:

```
public int f(int x[], int n)
   {
   int i, k = 0;
   if (n <= 0) return -1;
   i = 1;
   while (i < n)
      {
      if (x[k] > x[i])
          k = i;
      i = i+1;
      }
   return k;
   }
```

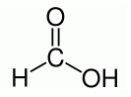
(a) What does the method **f** calculate?

the smallest index where the array has its minimal value

(b) What does it give as result if all fields of the array **x** contain the same number, let us say, 1 ?

0

Slide 109, tasks 1-3

1. Write a turtle command sequence which generates a balls-and-sticks molecule model of formic acid (structural formula: 

).
The atoms shall be represented by spheres with different sizes and colours (depending on the chemical element), and the bonds by cylinders (**F(**...**)** command of the turtle). The double bond shall be represented by a thicker cylinder.
Hint: The colour of an atom **X** can be specified as in the following example:

```
module X extends Sphere(1.0)
   {{ setShader(BLUE);}};
```

Test your solution with GroIMP.

```
module H extends Sphere(1.0)
   {{ setShader(WHITE);}};

module C extends Sphere(2.0)
   {{ setShader(BLACK);}};

module O extends Sphere(2.5)
   {{ setShader(RED);}};

protected void init()
   [
   Axiom ==> RU(180) O D(1) F(6) C D(0.5)
             [ RU(60) F(4) H ] RU(-60) F(6) O F(4) H;
   ]
```

2. (a) Write an L-system which simulates the primary growth of a plant in annual steps. The *annual shoots* of the vertical main axis (stem) shall all have the same length. The uppermost annual shoot shall bear an *apical bud* (= a red sphere) and a *lateral bud* (= a green sphere). The apical bud is supposed to produce a new annual shoot of the main axis next year, and from the lateral bud shall grow a shorter *lateral shoot* with a branching angle of 45°, which will terminate its growth next year (i.e., there are no buds at the lateral shoots). The positions of the lateral branches are alternating (left-right-left-right-...) along the stem. The simulation shall start with an apical bud.

```
module ApicalBud extends Sphere(1).(setShader(RED));
module LateralBud extends Sphere(0.5).(setShader(GREEN));
protected void init()
   [
   Axiom ==> D(0.5) ApicalBud;   /* simulation starts with apical bud */
   ]
public void grow()
   [
   ApicalBud ==> F(10) RH(180) [ M(-2) RU(45) LateralBud ]
                 ApicalBud;
                 /* F(10): annual shoot of main stem, all have same length
                    RH(180): provides alternating direction of branches
                    [ ... ]: lateral shoot
                    M(-2): position of lateral shoot a bit below top of mother shoot
                    RU(45): branching angle of 45°  */
   LateralBud ==> F(6); /* does not produce new LateralBud after shoot F(6)*/
   ]
```

(b) Modify the model by introducing a trend: Assume that the annual shoots get 10 per cent shorter each year.

We can use the turtle command **F0** instead of **F(10)**, thus letting the actual step length of the turtle (and thus the length of the produced shoots) depend on the current state of the turtle. This state can be modified by the command **LMul(***x***)**, where *x* is the reduction factor. Instead of absolute positioning of the branch by **M(–2)**, we should then use relative positioning by **MRel(0.8)**, placing the branch at a point at 80% of the mother shoot's length from the base.
Exchange the blocks of rules by the following (new parts are green):

```
protected void init()
   [
   Axiom ==> D(0.5) L(10) ApicalBud;
   ]

public void grow()
   [
   ApicalBud ==> F0 LMul(0.9) RH(180)
                 [ MRel(0.8) RU(45) LMul(0.6) LateralBud ]
                 ApicalBud;
   LateralBud ==> F0;
   ]
```

(c) Assume additionally that the apical bud produces a flower (= a large blue cone) after 7 years, and that the plant then stops to grow.

Test your solutions with GroIMP.

Remarks: By **`M(-`***s*`)`** you can cause the turtle to move back along the main axis by stepsize *s*. **`Cone(`***h*`, `*r*`)`** stands for a cone with height *h* and radius *r*.

```
module ApicalBud(int age) extends Sphere(1).(setShader(RED));
module LateralBud extends Sphere(0.5).(setShader(GREEN));
module Flower extends Cone(5, 6).(setShader(BLUE));

protected void init()
   [
   Axiom ==> D(0.5) L(10) ApicalBud(0);
   ]

public void grow()
   [
   ApicalBud(t), (t < 7) ==> F0 LMul(0.9) RH(180)
                             [ MRel(0.8) RU(45) LMul(0.6) LateralBud ]
                             ApicalBud(t+1);
   ApicalBud(t), (t == 7) ==> M(5) RU(180) Flower;
   LateralBud ==> F0;
   ]
```

3. Open the example "Molecules" in GroIMP's built-in example portfolio ("File" / "Show Examples").
Make several model runs by clicking on the buttons "Run run" and "Reset", and observe what happens.
Now modify the model in the following ways:
(a) Increase the number of atoms from 10 to 20.
(b) Switch off the output of text to the console.
(c) Double the distance threshold for formation of a bond between two atoms.

(a) modify line 26 of the code of the example,
(b) modify line 78,
(c) modify line 75.