

skip this level and the next if you just cannot wait to give and take life. come back when you need to know.

know that you may structure your sorkariak's genes into chromosomes that are like functions that may be invoked from anywhere, including from loops.

chromosome: a division of the genes* bounded by markers* and invoked* by name or by number.

invoke: send the program counter* to the first gene* of a chromosome*.

main chromosome: chromosome* number zero, invoked* as an infinite loop as cells* are born*.

marker: a gene* marking the end of a chromosome*.

program counter: the number of the next gene* to execute.

CHROMOSOMES 1

instruction (3), for instance, invokes the third chromosome. when a cell executes this instruction, it pushes the program counter on the stack and jumps to the first gene of the invoked chromosome.

instruction --- marks the end of a chromosome. when a cell executes it, it pops from the stack the program counter that was pushed when the chromosome was invoked and jumps to the gene that immediately follows it.

the main chromosome is numbered zero. when the cell reaches the end of the main chromosome, it jumps back to its first gene. all cells begin life pointing to this first gene, whether they have more than one chromosome or not. thus, a cell may be seen as an infinite loop on its main chromosome.

the following code steps 500 times before running toward its nearest neighbouring cell:

```
500 repeat step
infinity repeat (1)
---
```

approach step

this code cannot be written without chromosomes.

code the standard approach, approach step, followed by a conditional block that invokes (1) infinitely if proximity is smaller than 20. chromosome 1 steps 50 times and then turns.

you will see your cells go toward their neighbours and, when close enough, turn around each other until they disappear.

the main loop extends to the last gene if the program consists of a single chromosome. otherwise, it extends to the last gene of the first chromosome, before the first marker. (V:4)

a program may be divided in functions or subroutines. each of these functions, including the infinite loop of the main function, is called a chromosome. (VI:1)

chromosomes are numbered from zero to $n - 1$, where n is the number of chromosomes. (VI:2)

when the program counter reaches the marker at the end of a chromosome, it goes back to the gene immediately following the gene that invoked it, the calling gene's number being stored on the stack. (VI:6)