This is an example of an algorithm for the Control Program

START

DECLARE STRING yesOrNo and INITILIZE to "y"

DECLARE RatStatusInterface currentRatFacts and INITILIZE to NOTHING

DECLARE INTEGER numberOfRats and INITILIZE to 0

DECLARE INTEGER currentRatState and INITILIZE to 0

DECLARE String[] ratIDList and INITILIZE to NOTHING

DECLARE DesertInterface desert and INITILIZE to NOTHING

DECLARE BOOLEAN notFinished and INITILIZE to TRUE

DECLARE INTEGER index and INITILIZE to 0

REPEAT

{

SET notFinished EQUAL TO TRUE

SET desert EQUAL TO the REFERENCE TO a NEW Desert OBJECT USING Desert()

DISPLAY "How may rats would you like to run through the desert?"

SET numberOfRats EQUAL TO the RETURN VALUE FROM the CALL OF the METHOD Support.getBoundedIntegerValue(1, 20)

SET ratIDList EQUAL TO the REFERENCE TO a NEW ARRAY String [numberOfRats]

SET index EQUAL TO 0

WHILE index IS LESS THAN numberOfRats

SET ratIDList[index] TO the REFERENCE RETURNED FROM the method desert.startRat().getRatID()

SET index EQUAL TO index PLUS 1

END WHILE

WHILE notFinished IS TRUE run the rats across the desert

SET index EQUAL TO 0

WHILE index IS LESS THAN numberOfRats move each rat one cell By

SET currentRatFacts EQUAL TO the value returned from the CALL of the METHOD desert.moveRat(ratIDs[index])

SET currentRatState EQUAL TO the value returned from the CALL of the METHOD currentRatFacts.getRatState()

SWITCH on the value of currentRatState

CASE 1 rat is dead

SET ratIDList[index] EQUAL TO the value returned from the CALL of the METHOD desert.startRat().getRatID()

CASE 0 rat is waiting for next turn

Do nothing

END CASE 1 and 0

CASE -1

SET notFinished EQUAL TO FALSE to stop the outer WHILE loop

SET index EQUAL TO numberOfRats to stop this WHILE LOOP

END CASE -1

END SWITCH

END WHILE

END WHILE

CALL the METHOD in the desert to DIPLAY the route of the successful Rat and the numbers of dead rats.

DISPLAY "Would you like to run another Desert."

REPEATE

DSPLAY "Please enter \"y\" for yes or \"n\" for no"

SET yesOrNo EQUAL TO the value returned by the user

UNTIL the user enters “y" OR "n"

UNTIL yesOrNo IS EQUAL TO "n"