

AREA PROGRAM:

```
Disp "1-Rectangle"
Disp "2-Circle"
Disp "3-Triangle"

Input ("DETERMINE AREA FOR #",N)

If N=1
Then
Input ("LENGTH: ",L)
Input ("WIDTH: ",W)
L*W→A

Else
If N=2
Then
Input ("RADIUS: ",R)
 $\pi$ *R^2→A

Else
Input ("LENGTH: ",L)
Input ("HEIGHT: ",H)
(L*H)/2→A
End
End
Disp "AREA: "+toString(A)
```

RECURSIVE SEQUENCE PROGRAM:

```
ClrHome
Disp "NUMBER OF SEATS"
Disp "AT N TABLES"

Prompt N
ClrList L1,L2

4→S
1→L1(1)
S→L2(1)

If N>1
Then
For(T,2,N)
T→L1(T)
S+2→S
S→L2(T)
End
End

Disp "NUMBER OF TABLES: ",N
Disp "NUMBER OF SEATS: ",S
```

Lucky "Eu"

```
ClrHome
Disp "CAN YOU GUESS EULER'S "
Disp "LUCKY NUMBERS?"
Prompt N

1→A
0→Q
"is it lucky"
While A≤N-1 and Q=0

A^2-A+N→P
"is it prime?"

For(M,2,P-1)
If int(P/M)=P/M
Then
1→Q
End
End
A+1→A
End

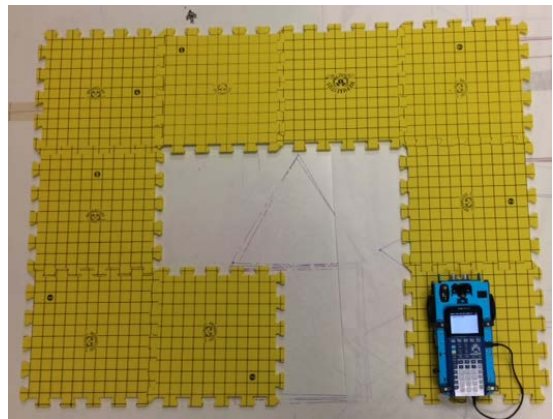
If Q=0 and N>1
Then
Disp "YES, "+toString(N)+" IS LUCKY"
Else
Disp "NO, "+toString(N)+" ISN'T LUCKY"
End
```

Coding with TI-Innovator Rover:

LEFT TURN PROGRAM:

```
Send("CONNECT RV")
Send("RV FORWARD 0.5 M")
Send("RV LEFT ")
Send("RV FORWARD 0.5 M")
```

NAVIGATION CHALLENGE : Use the FORWARD and LEFT commands to have rover navigate the course setup below. For simplicity sake, the distances required are 0.5 M, 0.85 M, 0.52 M and 0.3 M respectively.



Square Root "Newt"

```
ClrHome
Disp "Determine Square Roots"
Disp "Numerically. Input A,"
Disp "returns Sqrt(A), and "
Disp "iterations required"
Disp "based on the initial"
Disp "guess, G."

Prompt A,G
1→I

(1/2)(G+A/G)→R

While R≠V(A)
(1/2)(R+A/R)→R
1+I→I
Disp R
Disp I
Wait .5
End

ClrHome
Disp "approx root is"
Disp R
Disp "with "+toString(I)+" iterations"
```

Approximate e

```
ClrHome
Disp "THIS PROGRAM APPROXIMATES"
Disp "e BY SUMMING 1/(n!). "
Disp "ENTER A POSITIVE NUMBER "
Disp "FOR N AND SEE HOW QUICKLY"
Disp "THE SUM APPROACHES e."
Disp ""
Prompt N

1→E
If N≥1
Then
For(M,1,N)
E+1/M!→E
Disp E
Wait .5
End
Else
Disp "N MUST BE POSITIVE"
End
```

Random Sample from L1

```
dim(L1)→A
Input ("Sample Size: ",B)
Input ("With Repeats(Y/N): ",Str1)
If Str1="Y"
Then
ClrList L3,L2
For(C,1,B)
L1(randInt(1,A))→L2(C)
End
ClrHome
Disp "Sample stored in L2"
Disp L2
Else
ClrList L3,L2
randIntNoRep(1,A)→L3
For(C,1,B)
L1(L3(C))→L2(C)
End
ClrList L3
ClrHome
Disp "Sample stored in L2"
Disp L2
End
```