Penrose Encoding and graph visualization.

Diana Itzel Vázquez Santiago.
Emmanuel Isaac Juárez Caballero.
Jesús Eduardo Hermosilla Díaz.
Maestría en Inteligencia Artificial, generación 2021-2023

```
(*Starting Executing Cells*)

Clear["Global*"];

SetDirectory[NotebookDirectory[]];

FindExternalEvaluators["Python"]

session = StartExternalSession["Python"]

u = Import["nuniversal.txt"]//ToExpression;

n1 = 45081370446156395898211377564343790 8;

n2 = 10389728107;

n = u;

ExternalEvaluate[session, File["penrose_encoding.py"]];

PenroseEncoding = ExternalEvaluate[session, "codificacion"];

execution = PenroseEncoding[n2];
```

```
TM after Penrose Coding: R10R1R100R111RR1H
 Instructions from TM
0 00 -> 00R
1 01 -> 10R
2 10 -> 01R
3 11 -> 100R
4 100 -> 111R
5 101 -> 00R
6 110 -> 01H
```

```
MT = Import["universal.txt"];
```

```
MT = StringDelete[StringSplit[Text[MT][[1]], “,”], {“[”, “]”, “”, “ ”}];
```

MultiGraph2 implementation: A tool to visualize correctly multilabel graphs on Wolfram Mathematica

```
ClearAll[multiGraph2]

multiGraph2[vl_, elist_, elabels_, estyles_, o : OptionsPattern[Graph]]:=

Module[{esf, edges, labels, styles, sorted = Transpose@SortBy[Transpose[{elist, elabels, estyles}], {PositionIndex

{edges, labels, styles} = {sorted[[1]], ##&@@(RotateRight/@sorted[[2;;]])};

esf =

{First[styles = RotateLeft[styles]], GraphElementData[“Arrow”, “ArrowSize”->0.01][##]/.

Arrowheads[ah_]:>Arrowheads[Append[ah, {.0003, .7, Graphics[Text[Framed[First[labels = RotateLeft[labels]],

Graph[vl, edges, EdgeShapeFunction->esf, o]]
```

Table generation for associations in the graph in order to be functional with multigraph2

```
fn = ParallelTable[{{FromDigits[(StringSplit[MT[[i]], “->”][[1]]//Characters)[[1;; − 2]]//StringJoin, 2]},

{FromDigits[((StringSplit[MT[[i]], “->”][[1]]//Characters)[[1;; − 2]]//StringJoin), 2]

FromDigits[((StringSplit[MT[[i]], “->”][[2]]//Characters)[[;; − 3]]//StringJoin), 2]},

{StringJoin[(StringSplit[MT[[i]], “->”][[1]]//Characters)//Last, (StringSplit[MT[[i]], “->”][[2]]//Characters)[[−

(StringSplit[MT[[i]], “->”][[2]]//Characters)//Last]}}, {i, 1, Length[MT]}];
```

Visualization of the graph using Gravity Embedding, a method to visualize more clearly the graph.

```
styles = ColorData[97]/@Range[Length[fn[[;;, 3]]]];

g = multiGraph2[Flatten@Gather[(fn[[;;, 1]]//Flatten)//DeleteDuplicates], fn[[;;, 2]]//Flatten, fn[[;;, 3]]//Flatte

VertexLabels->Placed[“Name”, Center], ImageSize->Large, GraphLayout->“GravityEmbedding”]
```

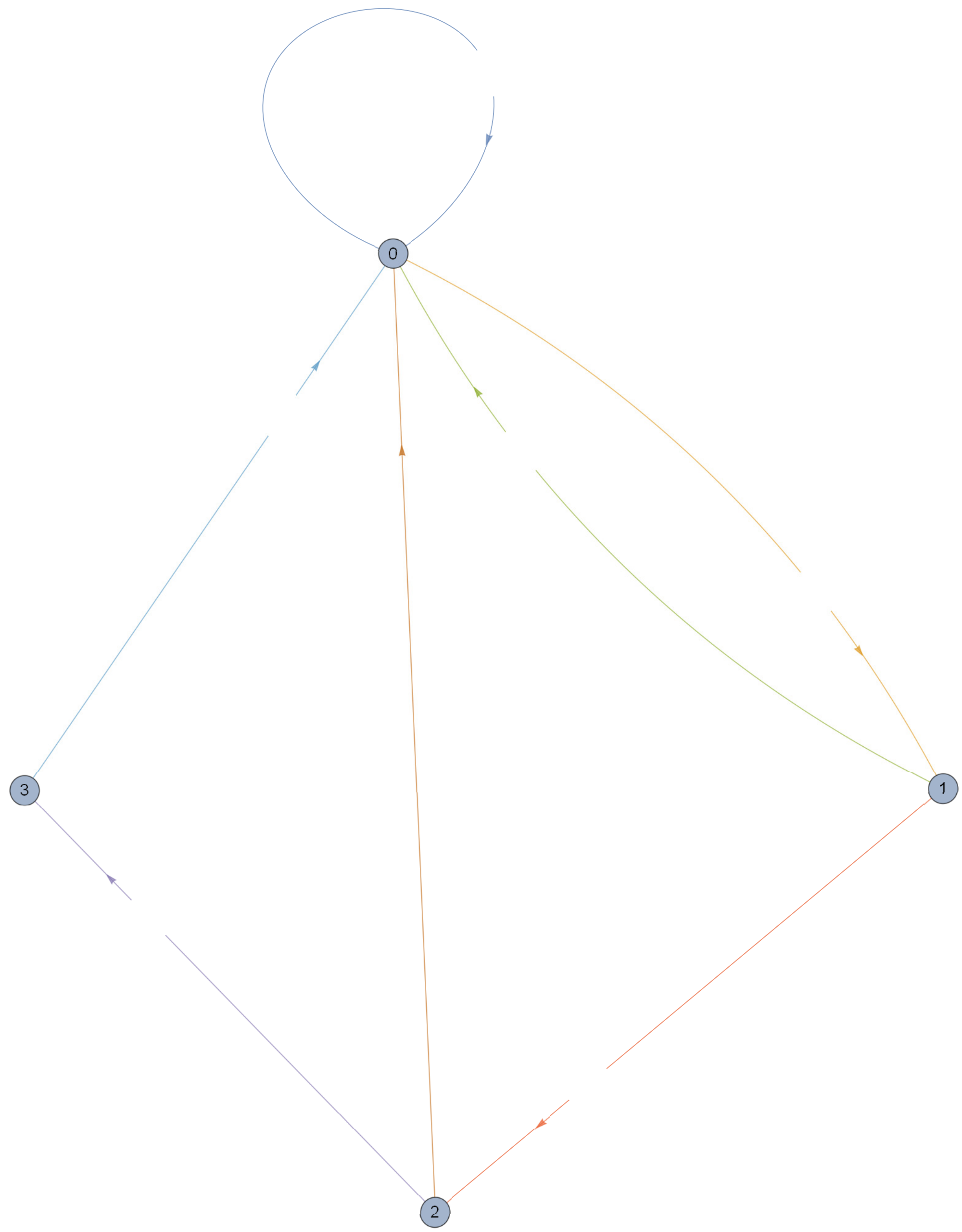
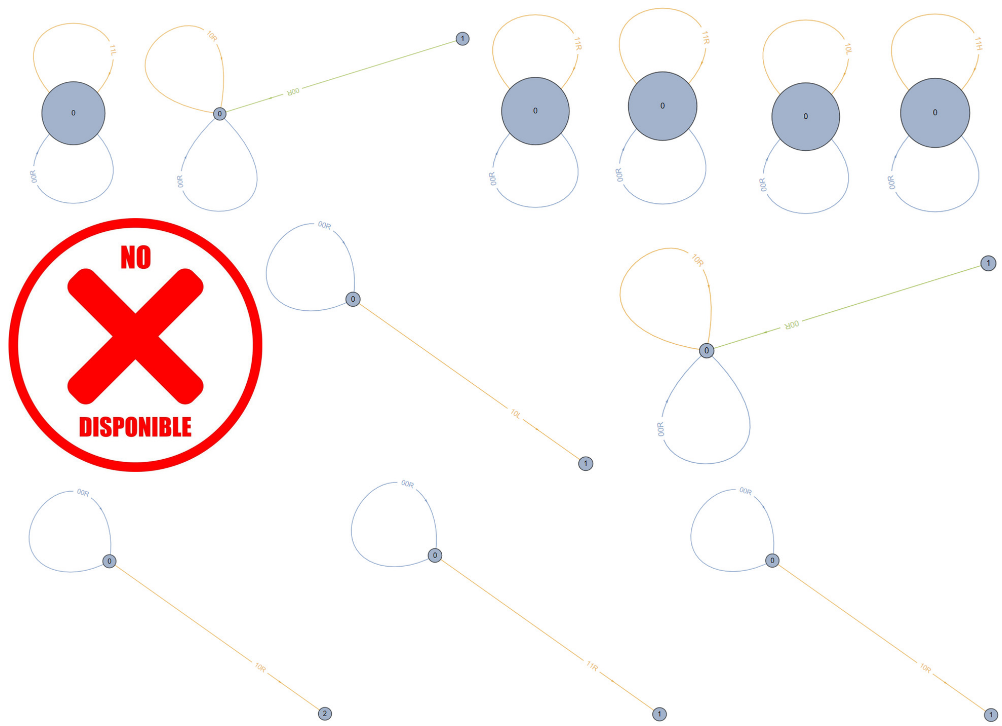


Image List for the first 12 Turing Machines

imageList = {, , , , , , , , , , , };

Visualization of first 12 TM

```
ImageCollage[imageList]
```




```
procesado = Table[{(*Estado inicial*)(StringSplit[MT[[i]], "->"][[1]]//Characters)[[1;; - 2]]//StringJoin//ToEx
(*Valor que leo*)(StringSplit[MT[[i]], "->"][[1]]//Characters)//Last//ToExpression,
(*Valor que escribo*)(StringSplit[MT[[i]], "->"][[2]]//Characters)[[-2]]//ToExpression,
(*Movimiento*)(StringSplit[MT[[i]], "->"][[2]]//Characters)//Last//ToExpression,
(*Estado siguiente*)(StringSplit[MT[[i]], "->"][[2]]//Characters)[[;; - 3]]//StringJoin//ToExpression
},
{i, 1, Length[MT]}];
Export["procesado.csv", procesado]
```

procesado.csv

In order to execute the decoder for the UTM we need to introduce a number to execute $U(n, m) =$

ExternalEvaluate[session, File["decoderTM.py"]];

TMexecution = ExternalEvaluate[session, "execution"];

execution = TMexecution[231];

0 \t 0 [1] 0 1 0 1 0 0 0 1 0 1 0 1 0 1 1 0
1 \t 0 0 [0] 1 0 1 0 0 0 1 0 1 0 1 0 1 1 0
0 \t 0 0 1 [1] 0 1 0 0 0 1 0 1 0 1 0 1 1 0
1 \t 0 0 1 0 [0] 1 0 0 0 1 0 1 0 1 0 1 1 0
0 \t 0 0 1 0 1 [1] 0 0 0 1 0 1 0 1 0 1 1 0
1 \t 0 0 1 0 1 0 [0] 0 0 1 0 1 0 1 0 1 1 0
0 \t 0 0 1 0 1 0 1 [0] 0 1 0 1 0 1 0 1 1 0
0 \t 0 0 1 0 1 0 1 0 [0] 1 0 1 0 1 0 1 1 0
0 \t 0 0 1 0 1 0 1 0 0 [1] 0 1 0 1 0 1 1 0
1 \t 0 0 1 0 1 0 1 0 0 0 [0] 1 0 1 0 1 1 0
0 \t 0 0 1 0 1 0 1 0 0 0 1 [1] 0 1 0 1 1 0
1 \t 0 0 1 0 1 0 1 0 0 0 1 0 [0] 1 0 1 1 0
0 \t 0 0 1 0 1 0 1 0 0 0 1 0 1 [1] 0 1 1 0
1 \t 0 0 1 0 1 0 1 0 0 0 1 0 1 0 [0] 1 1 0
0 \t 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1 [1] 1 0
1 \t 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1 0 [1] 0
10 \t 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1 0 0 [0]
11 \t 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1 0 0 1 [0]
H \t 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1 0 0 1 1 [0]

Formato posible para la visualización de las máquinas de Turing.

a = {{0, "[1]", 0, 0, 0, 1, 1, 0},

{0, 0, "[0]", 0, 0, 1, 1, 0},

{0, 0, 1, "[0]", 0, 1, 1, 0},

{0, 0, 1, 0, "[0]", 1, 1, 0},

{0, 0, 1, 0, 0, "[1]", 1, 0},

{0, 0, 1, 0, 0, 0, "[1]", 0},

{0, 0, 1, 0, 0, 0, 0, "[0]"},

{0, 0, 1, 0, 0, 0, 0, 1, "[0]"},

{0, 0, 1, 0, 0, 0, 0, 1, 1, “[0]”}

};

a/.{“[0]”->Pink, “[1]”->Green, 0->Yellow, 1->Yellow}//Grid

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □ □ □