4x4x4 Cube Synthesizer

Preliminary Document

Introduction

Revenge Cubes are **4x4x4** cubes. They may be solved for a *single* face out of six to display pictures or patterns on a selected front face.

A way of solving a single face is first to solve the 4 centers, then the 4 dedges (double edges) and finally the 4 remaining corners cubes. This is very similar to the method used for solving the *first* layer of a regular Revenge Cube.

By twisting and rotating some parts of the cube, it is possible to move selected cubelets from any face to the front face. Note that each letter, number or symbol on a solved face should have the right orientation, i.e. should generally be oriented North (0°).

Maneuvers used for twisting and rotating parts of a cube are coded as a series of letters called an *algorithm*. An algorithm is then a code for a sequence of moves used to change the state of a cube from an *initial* (unsolved) state to a *final* (solved) state.

The *initial* state of a cube is given by a particular layout of letters, numbers and symbols shown on the cube *texture* whereas the *final* state is given by what we would like to see displayed on a selected front face. This is where we would need a software tool for *automatically* generating an algorithm to set the cube to a *user-selected* final state. This is what is called *synthesis*, which is just the reverse of *analysis*. and the software tool to do this is called a *Synthesizer*. The Synthesizer input data is the *final* state data.

Basic algorithms are used to change the state of each front face cubie from an initial to a final state. Synthesized algorithms are then basic algorithms that have been concatenated.

Download CubeSynthesizer4 Version 1.0
Microsoft Excel 2007
http://www.mementoslangues.fr/CubeDesign/CubeSynthesizer4.xlsm
Microsoft Excel 97-2003
http://www.mementoslangues.fr/CubeDesign/CubeSynthesizer4.xls

4x4x4 Demo Cubes		
Reference Cube	**Demo**	**Demo**

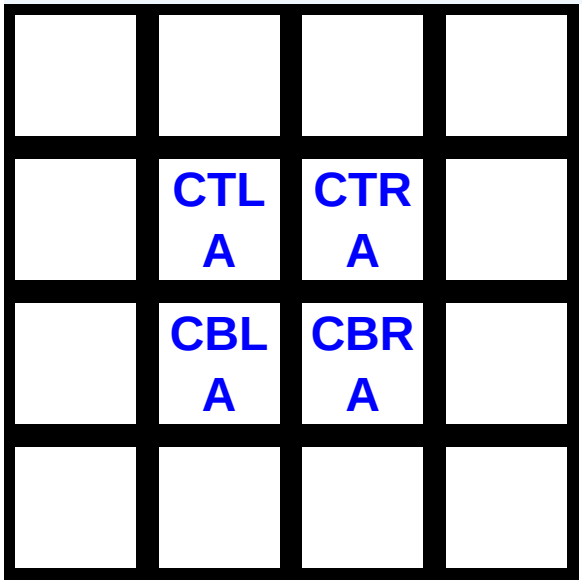
4x4x4 Cube Centers: Notations

Isolated (or strings of) letters, numbers, characters, symbols or pictures are placed on center cubies. These may be oriented **N**orth (0°), **E**ast (90°), **S**outh (180°) or **W**est (270°). Center cube notations are abbreviated as follows:

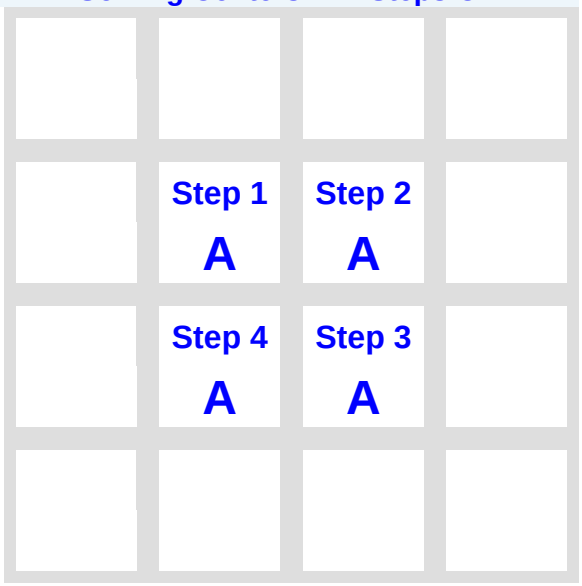
- **C**enter **T**op **L**eft (**CTL**)
- **C**enter **T**op **R**ight (**CTR**)
- **C**enter **B**ottom **L**eft (**CBL**)
- **C**enter **B**ottom **R**ight (**CBR**)

Notations & Solving Sequence

Center Cube Notations

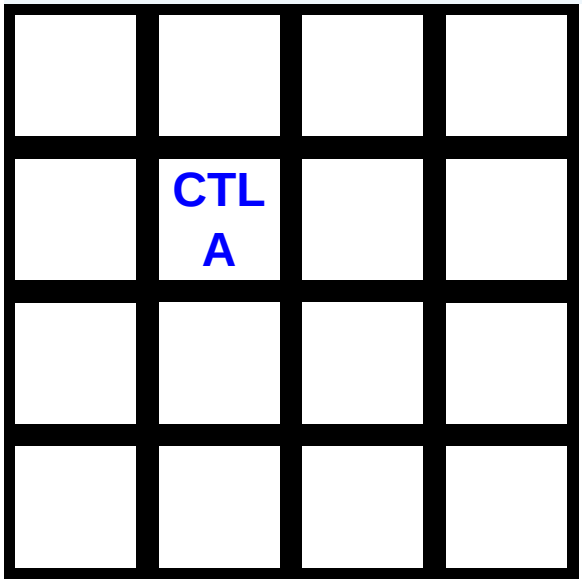


Solving Centers in 4 Steps CW

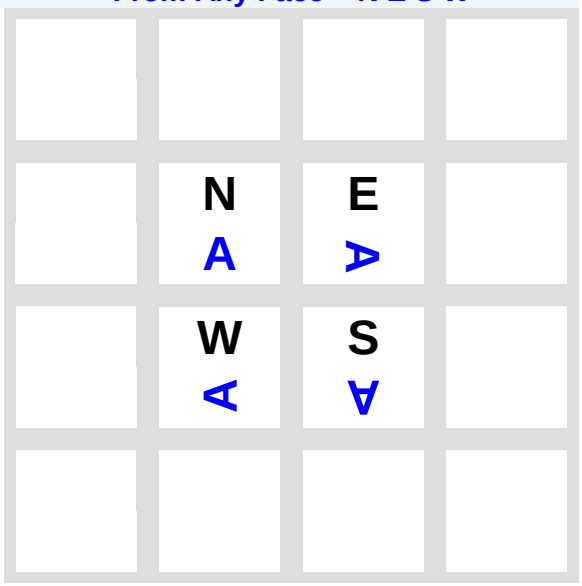


Center Cube Moves – Step 1

To Front Face – CTL North

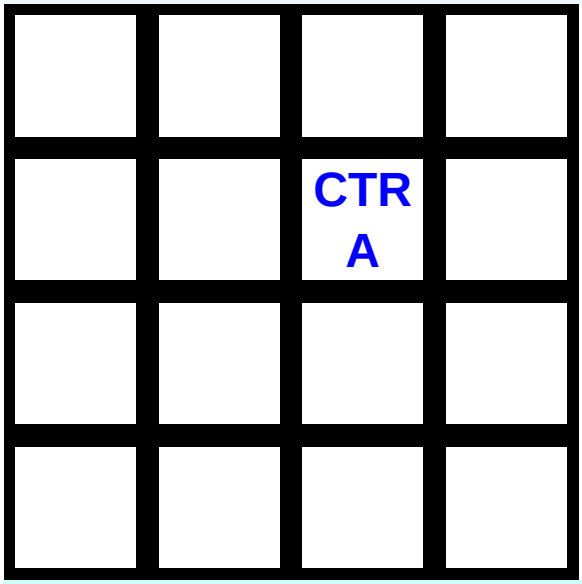


From Any Face – N E S W

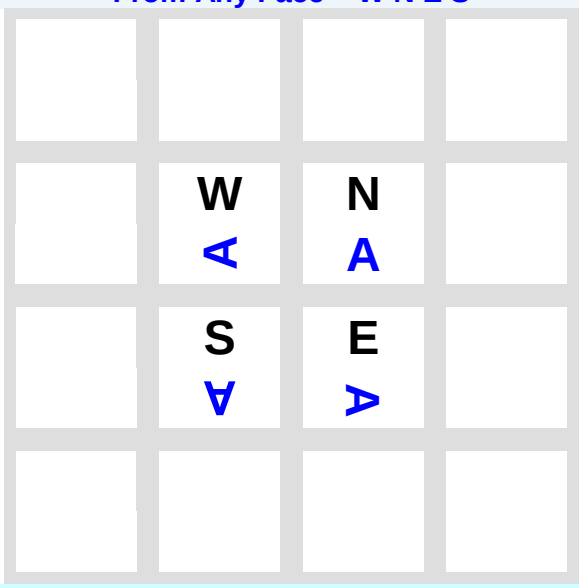


Center Cube Moves – Step 2

To Front Face – CTR North

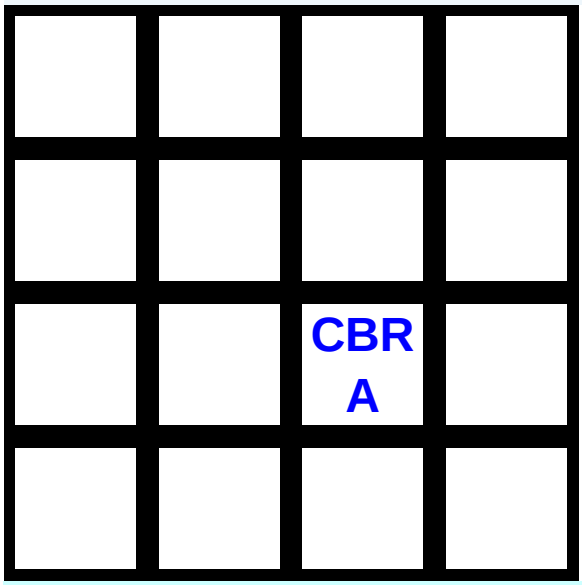


From Any Face – W N E S

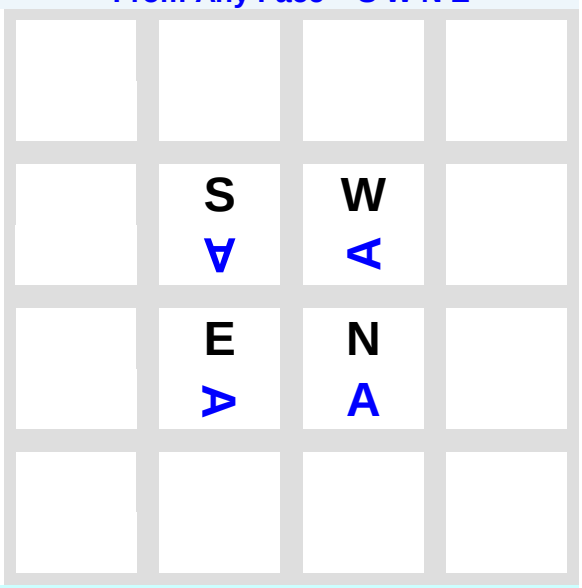


Center Cube Moves – Step 3

To Front Face – CBR North

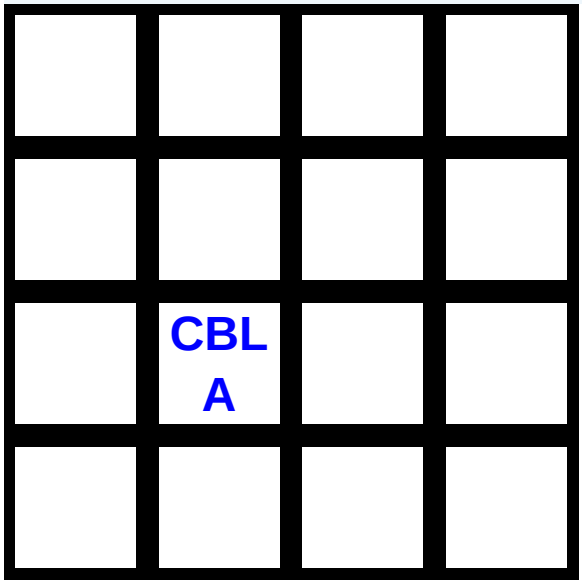


From Any Face – S W N E

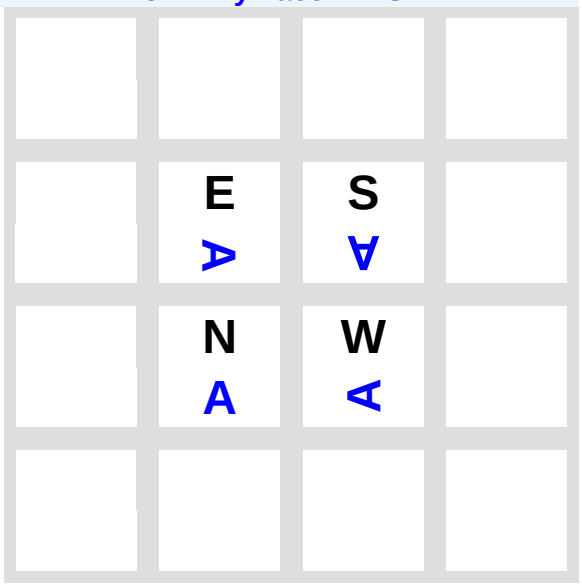


Center Cube Moves – Step 4

To Front Face – CBL North



From Any Face – E S W N



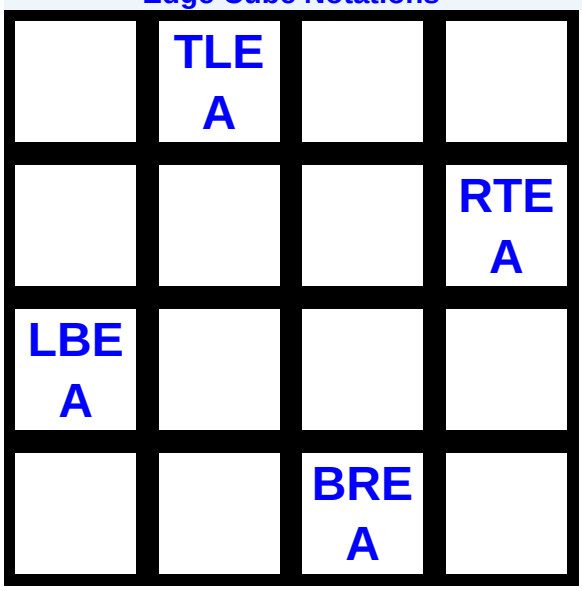
4x4x4 Cube Edges 1: Notations

Isolated (or strings of) letters, numbers, characters, symbols or pictures are placed on edge cubies. These may be oriented **N**orth (0°), **E**ast (90°), **S**outh (180°) or **W**est (270°). Edge cube notations are abbreviated as follows:

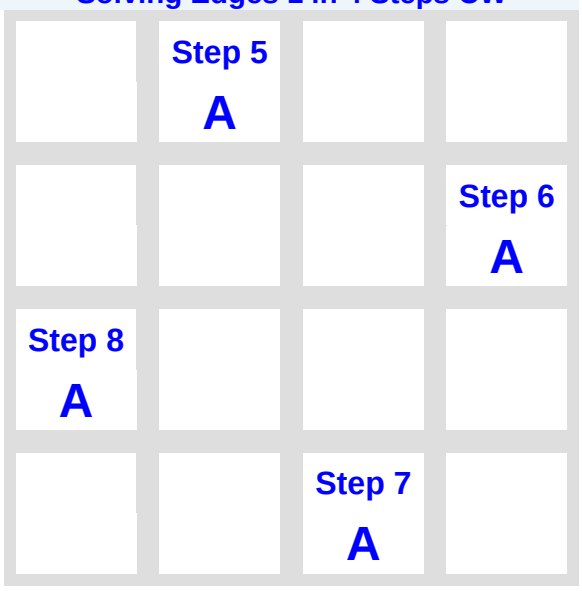
- **T**op **L**eft **E**dge (**TLE**)
- **R**ight **T**op **E**dge (**RTE**)
- **B**ottom **R**ight **E**dge (**BRE**)
- **L**eft **B**ottom **E**dge (**LBE**)

Notations & Solving Sequence

Edge Cube Notations

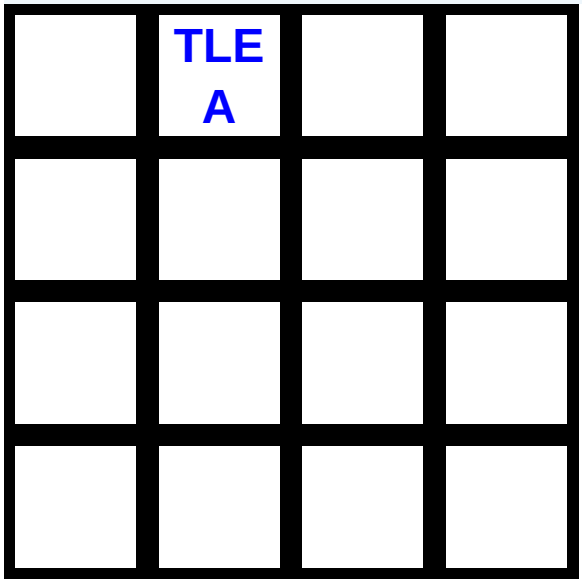


Solving Edges 1 in 4 Steps CW

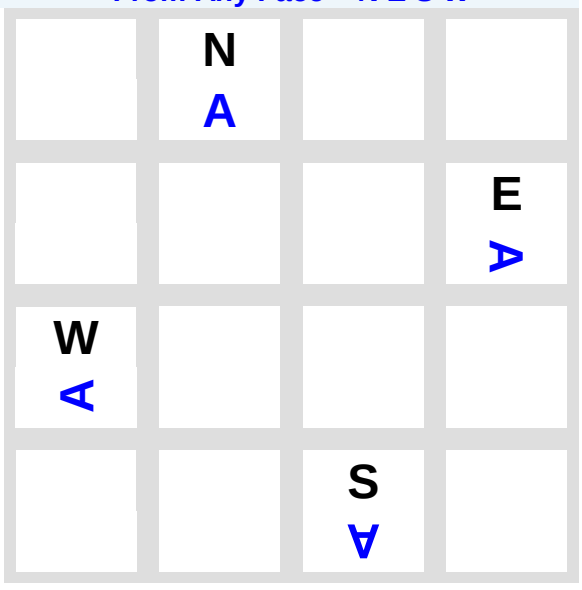


Edge Cube Moves – Step 5

To Front Face –TLE North

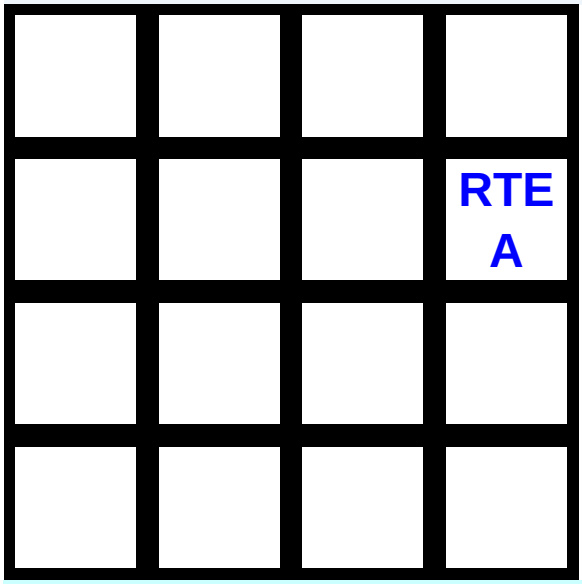


From Any Face – N E S W

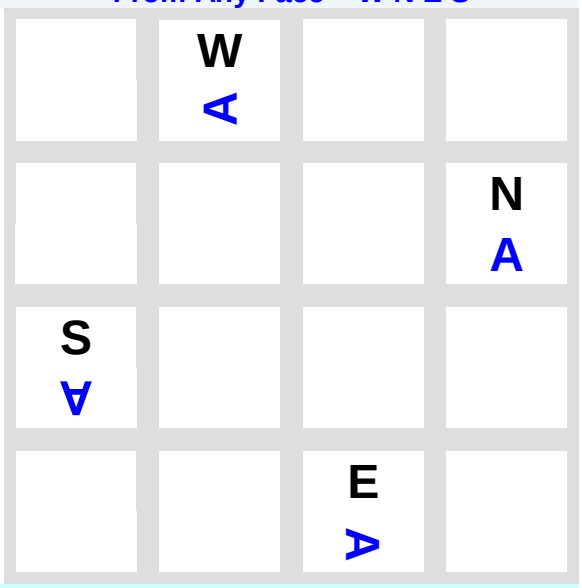


Edge Cube Moves – Step 6

To Front Face –RTE North

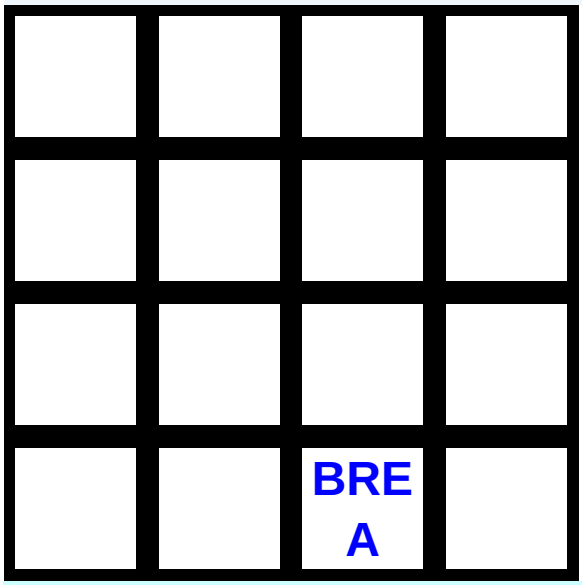


From Any Face – W N E S

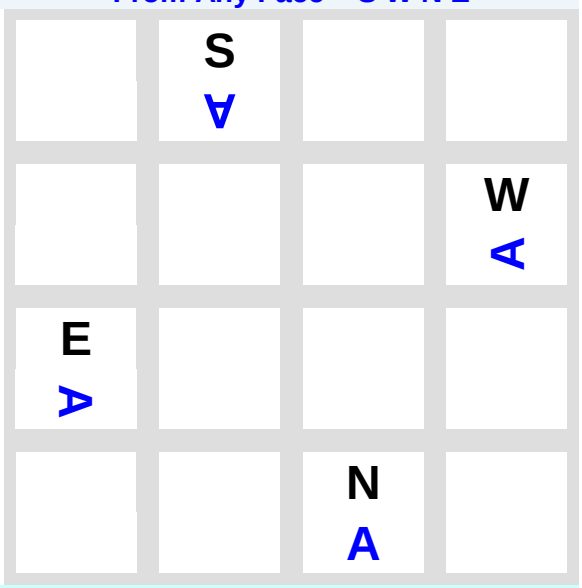


Edge Cube Moves – Step 7

To Front Face –BRE North

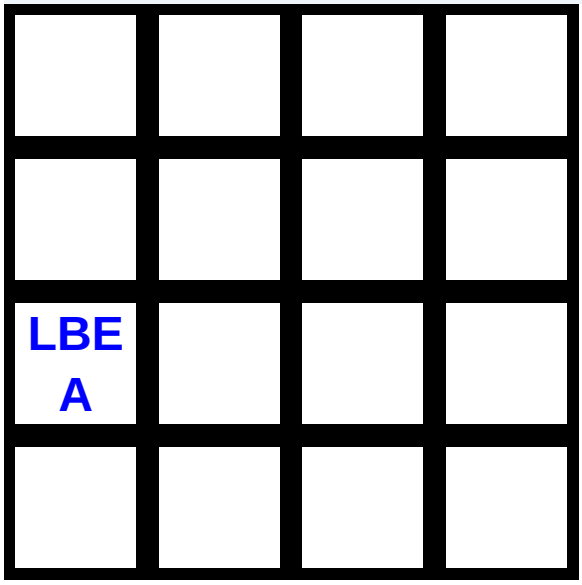


From Any Face – S W N E

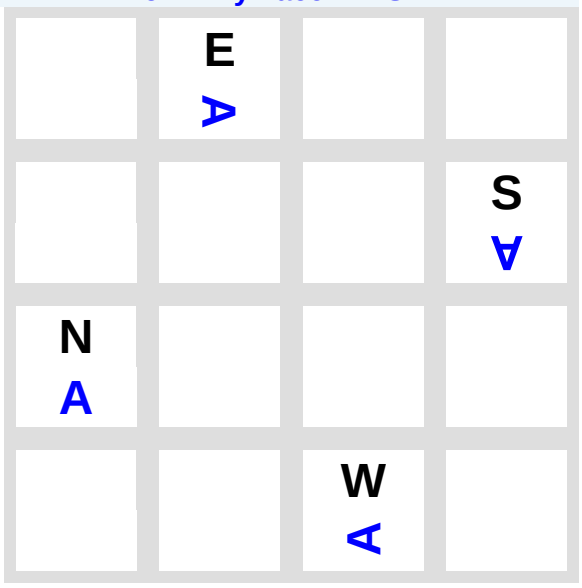


Edge Cube Moves – Step 8

To Front Face – LBE North



From Any Face – E S W N



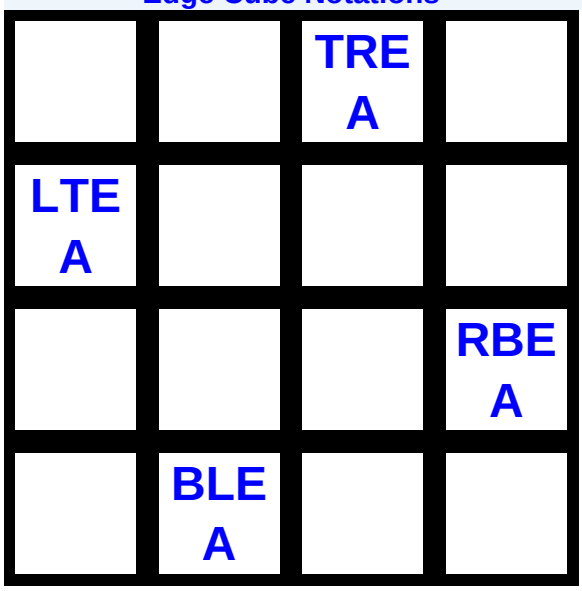
4x4x4 Cube Edges 2: Notations

Isolated (or strings of) letters, numbers, characters, symbols or pictures are placed on edge cubies. These may be oriented **N**orth (0°), **E**ast (90°), **S**outh (180°) or **W**est (270°). Edge cube notations are abbreviated as follows:

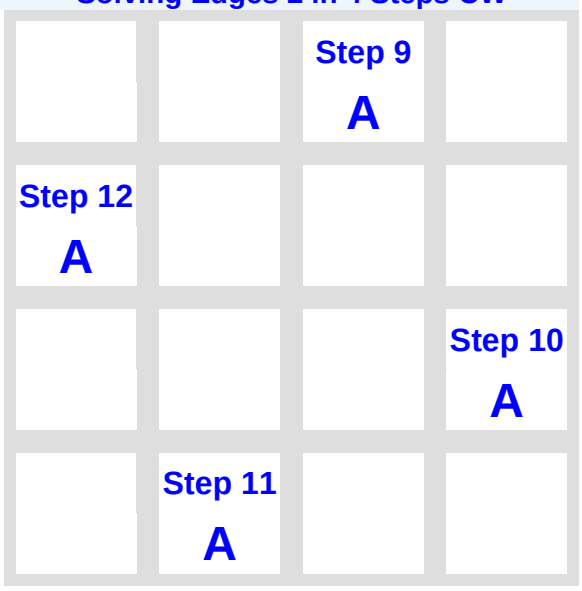
- **L**eft **T**op **E**dge (**LTE**)
- **L**eft **B**ottom **E**dge (**LBE**)
- **R**ight **T**op **E**dge (**RTE**)
- **R**ight **B**ottom **E**dge (**RBE**)

Notations & Solving Sequence

Edge Cube Notations

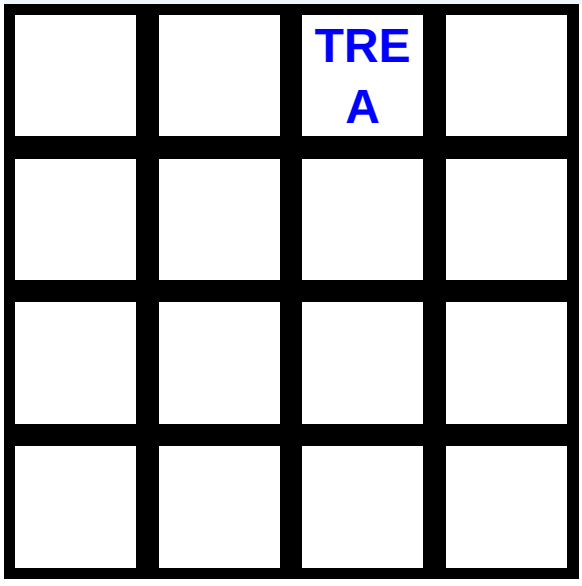


Solving Edges 2 in 4 Steps CW

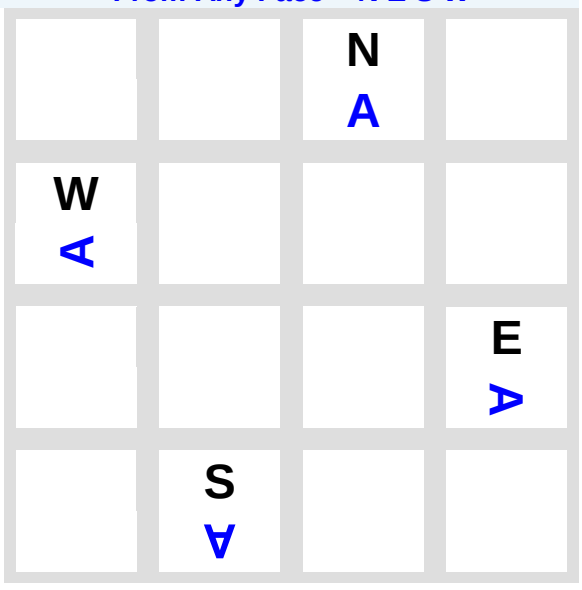


Edge Cube Moves – Step 9

To Front Face –TRE North

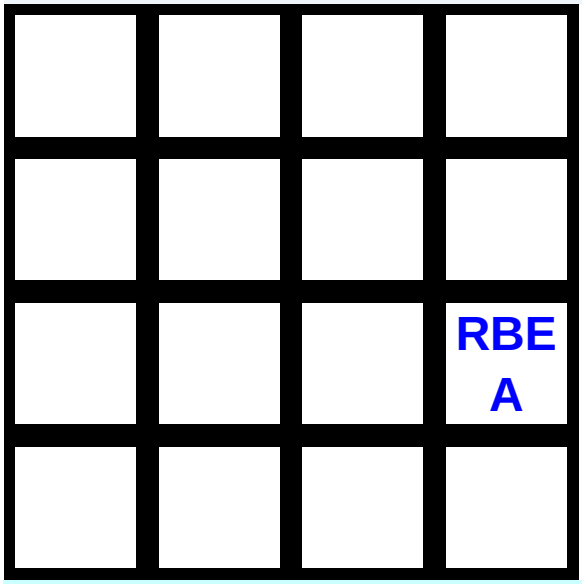


From Any Face – N E S W

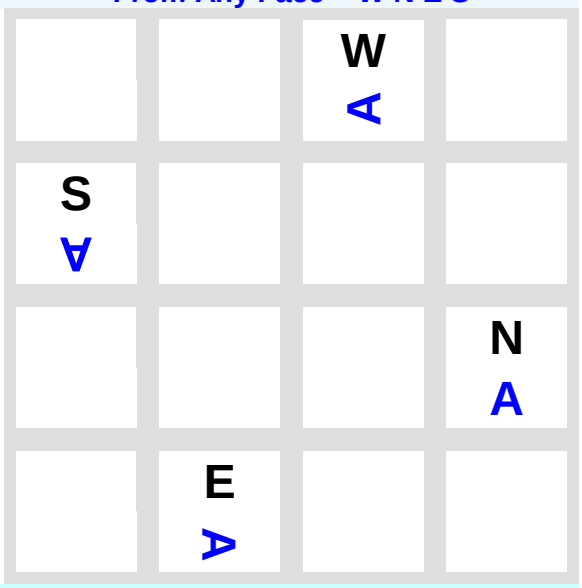


Edge Cube Moves – Step 10

To Front Face –RBE North

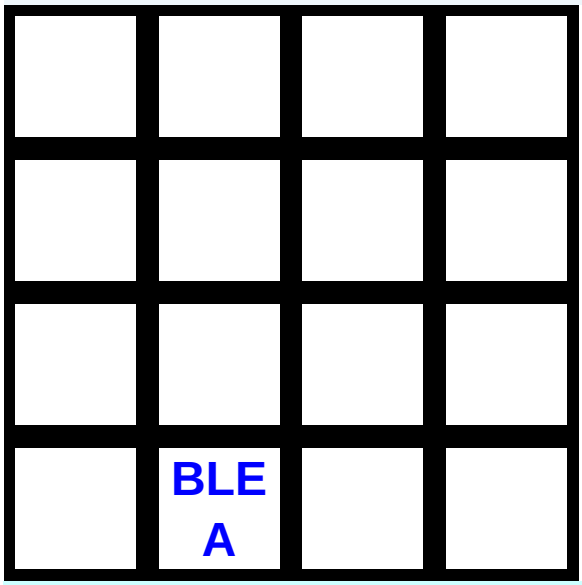


From Any Face – W N E S

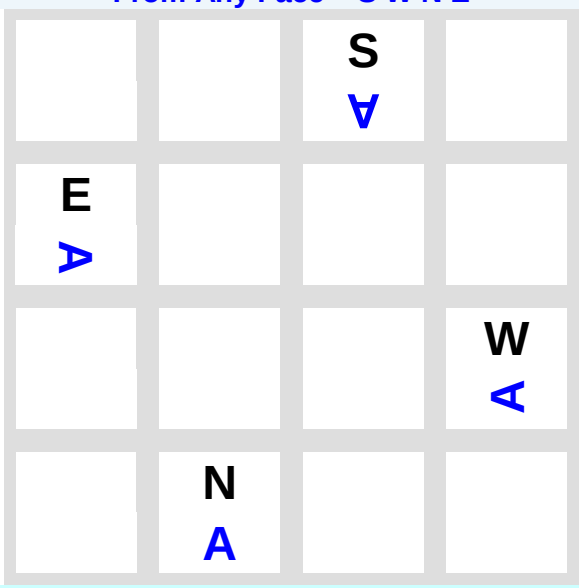


Edge Cube Moves – Step 11

To Front Face –BLE North

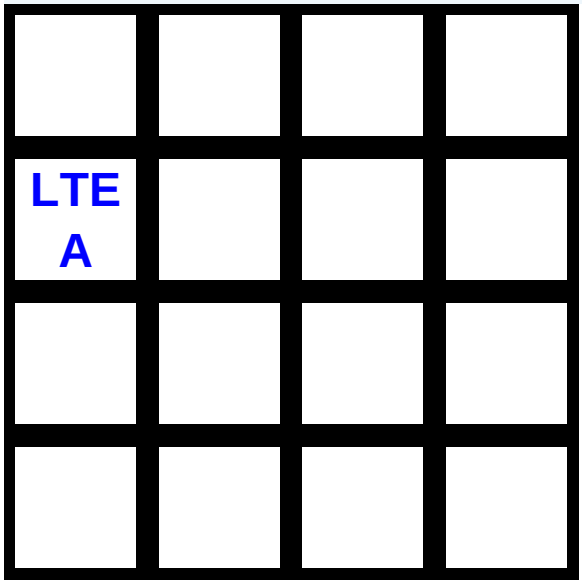


From Any Face – S W N E

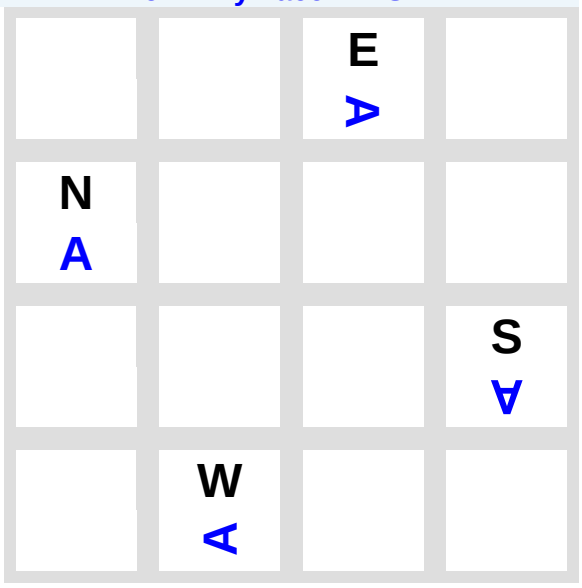


Edge Cube Moves – Step 12

To Front Face – LTE North



From Any Face – E S W N



4x4x4 Cube Corners: Notations

Isolated (or strings of) letters, numbers, characters, symbols or pictures are placed on corner cubies. These may be oriented **N**orth (0°), **E**ast (90°), **S**outh (180°) or **W**est (270°). Corner cube notations are abbreviated as follows:

- **T**op **L**eft **C**orner (**TLC**)
- **T**op **R**ight **C**orner (**TRC**)
- **B**ottom **L**eft **C**orner (**BLC**)
- **B**ottom **R**ight **C**orner (**BRC**)

Notations & Solving Sequence

Corner Cube Notations				Solving Corners in 4 Steps CW			
TLC A			TRC A	Step 13 A			Step 14 A
BLC A			BRC A	Step 16 A			Step 15 A

Corner Cube Moves – Step 13

To Front Face –TLC North				From Any Face – N E S W			
TLC A				N A			E A
				W A			S A

Corner Cube Moves – Step 14

To Front Face –TRC North				From Any Face – W N E S			
			TRC A	W A			N A
				S A			E A

Corner Cube Moves – Step 15

To Front Face –BRC North				From Any Face – S W N E			
				S A			W A
			BRC A	E A			N A

Corner Cube Moves – Step 16

To Front Face –BLC North				From Any Face – E S W N			
				E A			S A
BLC A				N A			W A

4x4x4 Cube: Notations

Cubies Location

				U	U	U	U									TLC	TLE	TRE	TRC				
				U	U	U	U									LTE	TLE	CTR	RTE				
				U	U	U	U									LBE	CBL	CBR	RBE				
				U	U	U	U									BLC	BLE	BRE	BRC				
L	L	L	L	F	F	F	F	R	R	R	R	TLC	TLE	TRE	TRC	TLC	TLE	TRE	TRC	TLC	TLE	TRE	TRC
L	L	L	L	F	F	F	F	R	R	R	R	LTE	TLE	CTR	RTE	LTE	TLE	CTR	RTE	LTE	TLE	CTR	RTE
L	L	L	L	F	F	F	F	R	R	R	R	LBE	CBL	CBR	RBE	LBE	CBL	CBR	RBE	LBE	CBL	CBR	RBE
L	L	L	L	F	F	F	F	R	R	R	R	BLC	BLE	BRE	BRC	BLC	BLE	BRE	BRC	BLC	BLE	BRE	BRC
				D	D	D	D	B	B	B	B					TLC	TLE	TRE	TRC	TLC	TLE	TRE	TRC
				D	D	D	D	B	B	B	B					LTE	TLE	CTR	RTE	LTE	TLE	CTR	RTE
				D	D	D	D	B	B	B	B					LBE	CBL	CBR	RBE	LBE	CBL	CBR	RBE
				D	D	D	D	B	B	B	B					BLC	BLE	BRE	BRC	BLC	BLE	BRE	BRC

Cubies Numbering and Orientation – Initial State (Texture State)

Intial State Numbering

Initial State Orientation (degrees)

				33	34	35	36									0	0	0	0				
				37	38	39	40									0	0	0	0				
				41	42	43	44									0	0	0	0				
				45	46	47	48									0	0	0	0				
49	50	51	52	1	2	3	4	17	18	19	20	0	0	0	0	0	0	0	0	0	0	0	0
53	54	55	56	5	6	7	8	21	22	23	24	0	0	0	0	0	0	0	0	0	0	0	0
57	58	59	60	9	10	11	12	25	26	27	28	0	0	0	0	0	0	0	0	0	0	0	0
61	62	63	64	13	14	15	16	29	30	31	32	0	0	0	0	0	0	0	0	0	0	0	0
				65	66	67	68	81	82	83	84					0	0	0	0	0	0	0	0
				69	70	71	72	85	86	87	88					0	0	0	0	0	0	0	0
				73	74	75	76	89	90	91	92					0	0	0	0	0	0	0	0
				77	78	79	80	93	94	95	96					0	0	0	0	0	0	0	0

Example: Center Cubes States – Step 1: From R [TLE, N] To F [TLE, N] To be completed

Numbering

				33	34	35	36				
				37	38	39	40				
				41	42	43	44				
				45	46	47	48				
49	50	51	52	1	2	3	4	17	18	19	20
53	54	55	56	5	6	7	8	21	22	23	24
57	58	59	60	9	10	11	12	25	26	27	28
61	62	63	64	13	14	15	16	29	30	31	32
				65	66	67	68	81	82	83	84
				69	70	71	72	85	86	87	88
				73	74	75	76	89	90	91	92
				77	78	79	80	93	94	95	96

Orientation (degrees)

				0	0	0	0				
				0	0	0	0				
				0	0	0	0				
				0	0	0	0				
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
				0	0	0	0	0	0	0	0
				0	0	0	0	0	0	0	0
				0	0	0	0	0	0	0	0
				0	0	0	0	0	0	0	0

Algorithm Synthesizer

Introduction

A computer program named *CubeSynthesizer4C* has been designed for synthesizing algorithms for moving center cubelets on 4x4x4 cubes that need to be solved for a *single* face. The program has been developed using Microsoft Office Excel and Visual Basic Editor. There is only a Developer's version of this program available at present.

The program can be used as follows:

1- Open *CubeSynthesizer4C* in Excel
2- Press Ctrl+Shift+S to display the Synthesizer Input Form
3- Select a cube from the Form
4- Click the OK Button and wait until algorithm synthesis is completed
5- Browse through the list of synthesized algorithms in Worksheet 'Main'
6- Copy a selected algorithm in CubeTwister or in an applet

Algorithms

A 4x4x4 Reference Cube has been used to check basic algorithms and to fill in lookup tables with numbers indicating the cube state. All basic algorithms are based on *commutators* and do not modify any facelet on the *Front* face other than the origin or destination facelets. A *complete* algorithm for a whole front face would then need up to 16 basic algorithms whereas a composite algorithm for the 4 centers is obtained by *concatenating* a maximum of 5 basic algorithms.

Synthesized algorithms are basic algorithms that have been concatenated. Finding basic algorithms can de done *manually* by searching the path of letter 'A' on a cube wire grid model, from an initial to a final location. The tip of Letter 'A' is used to show the orientation of a facelet.

Short Program description

There are 5 steps for solving the 4 centers on a front face. Step 0 is the first step, used to bring any of the 6 faces on front, with the correct orientation. Step 0 is simply a combination of cube rotations. Steps 1 to 4 are then applied CW (ClockWise), 1 step per center facelet. There is an option for optimizing the order of steps 1 to 4 to find the shortest length algorithm. In this case, algorithms are computed for 4! = 24 sequences and the shortest length algorithm is selected at the end of the optimization process. These sequences are shown in the Table below.

Algorithm Length Optimization – The 24 Sequences of Steps											
1	2	3	4	5	6	7	8	9	10	11	12
Step 1	Step 1	Step 1	Step 1	Step 1	Step 1	Step 2	Step 2	Step 3	Step 3	Step 4	Step 4
Step 2	Step 2	Step 3	Step 3	Step 4	Step 4	Step 1	Step 1	Step 1	Step 1	Step 1	Step 1
Step 3	Step 4	Step 2	Step 4	Step 2	Step 3	Step 3	Step 4	Step 2	Step 4	Step 2	Step 3
Step 4	Step 3	Step 4	Step 2	Step 3	Step 2	Step 4	Step 3	Step 4	Step 2	Step 3	Step 2
13	14	15	16	17	18	19	20	21	22	23	24
Step 2	Step 2	Step 2	Step 2	Step 3	Step 3	Step 3	Step 3	Step 4	Step 4	Step 4	Step 4
Step 3	Step 3	Step 4	Step 4	Step 4	Step 4	Step 2	Step 2	Step 2	Step 2	Step 3	Step 3
Step 4	Step 1	Step 1	Step 3	Step 1	Step 2	Step 4	Step 1	Step 1	Step 3	Step 1	Step 2
Step 1	Step 4	Step 3	Step 1	Step 2	Step 1	Step 1	Step 4	Step 3	Step 1	Step 2	Step 1

For each step, basic algorithms are automatically selected in look-up tables. A synthesized algorithm is then obtained by concatenating 5 basic algorithms. In order to shorten the synthesized algorithm length, *trivial* combinations between successive basic algorithms such as **B B'** or **MF MF'** are suppressed or simplified when concatenating basic algorithms.

Examples

Example #1: 4x4x4 Reference Cube

4x4x4 Reference Cube

Texture

```
                33 34 35 36
                37 38 39 40
                41 42 43 44
                45 46 47 48
49 50 51 52  1  2  3  4 17 18 19 20
53 54 55 56  5  6  7  8 21 22 23 24
57 58 59 60  9 10 11 12 25 26 27 28
61 62 63 64 13 14 15 16 29 30 31 32
            65 66 67 68 81 82 83 84
            69 70 71 72 85 86 87 88
            73 74 75 76 89 90 91 92
            77 78 79 80 93 94 95 96
```

Virtual Cube

4x4x4 Reference Cube – Basic Algorithms Examples

Step 1: From R [TLE, N] To F [TLE, N]

Number '**22**' is brought from face **R** to face **F** (+0°)

Step 2: From U [CBR, E] To F [CTR, N]

Number '**43**' is brought from face **U** to face **F** (+270°)

Basic Algorithms (Commutators)

MF' MU' MF MU

MU MF' MU' MF

A 4x4x4 Reference Cube is used to check basic algorithms and fill in lookup tables with numbers indicating the cube state. All basic algorithms do not modify any facelet on the *Front* face other than the origin or destination facelets.

Example #2: (Demo)

Example #3: (Demo)

R4 Cube Rotation Algorithms (Check: done)

Step 0 – R4 Cube Rotation Algorithms: From F/R/U/L/D/B to F [N]					
To F [N]		**From F**			
	Orientation		Orientation	Algorithm	Moves
	N		N	No move	0
	N		E	CF'	0
	N		S	CF2	0
	N		W	CF	0
To F [N]		**From R**			
	Orientation		Orientation	Algorithm	Moves
	N		N	CU	0
	N		E	CF' CR'	0
	N		S	CU' CR2	0
	N		W	CF CR	0
To F [N]		**From U**			
	Orientation		Orientation	Algorithm	Moves
	N		N	CR'	0
	N		E	CF' CU'	0
	N		S	CR CU2	0
	N		W	CF CU	0
To F [N]		**From L**			
	Orientation		Orientation	Algorithm	Moves
	N		N	CU'	0
	N		E	CF' CR	0
	N		S	CF2 CU	0
	N		W	CF CR'	0
To F [N]		**From D**			
	Orientation		Orientation	Algorithm	Moves
	N		N	CR	0
	N		E	CF' CU	0
	N		S	CF2 CR'	0
	N		W	CF CU'	0
To F [N]		**From B**			
	Orientation		Orientation	Algorithm	Moves
	N		N	CU2	0
	N		E	CF CU2	0
	N		S	CR2	0
	N		W	CF' CU2	0

R4 Edge 1 Move Algorithms (Check: undone)

Front face center cubies other than the origin and destination center cubies are left unchanged.

Step 5 – R4 Edge 1 Move Algorithms: From F/R/U/L/D/B to F [TLE, N]					
To F [TLE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLE	N	TLE	N	No move	0
TLE	N	RTE	E		
TLE	N	BRE	S		
TLE	N	LBE	W		
To F [TLE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLE	N	TLE	N		
TLE	N	RTE	E		
TLE	N	BRE	S		
TLE	N	LBE	W		
To F [TLE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLE	N	TLE	N		
TLE	N	RTE	E		
TLE	N	BRE	S		
TLE	N	LBE	W		
To F [TLE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLE	N	TLE	N		
TLE	N	RTE	E		
TLE	N	BRE	S		
TLE	N	LBE	W		
To F [TLE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLE	N	TLE	N		
TLE	N	RTE	E		
TLE	N	BRE	S		
TLE	N	LBE	W		
To F [TLE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLE	N	TLE	N		
TLE	N	RTE	E		
TLE	N	BRE	S		
TLE	N	LBE	W		

Step 6 – R4 Edge 1 Move Algorithms: From F/R/U/L/D/B to F [RTE, N]					
To F [RTE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RTE	N	TLE	W		0
RTE	N	RTE	N	No Move	
RTE	N	BRE	E		
RTE	N	LBE	S		
To F [RTE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RTE	N	TLE	W		0
RTE	N	RTE	N		
RTE	N	BRE	E		
RTE	N	LBE	S		
To F [RTE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RTE	N	TLE	W		
RTE	N	RTE	N		
RTE	N	BRE	E		
RTE	N	LBE	S		
To F [RTE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RTE	N	TLE	W		
RTE	N	RTE	N		
RTE	N	BRE	E		
RTE	N	LBE	S		
To F [RTE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RTE	N	TLE	W		
RTE	N	RTE	N		
RTE	N	BRE	E		
RTE	N	LBE	S		
To F [RTE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RTE	N	TLE	W		
RTE	N	RTE	N		
RTE	N	BRE	E		
RTE	N	LBE	S		

Step 7 – R4 Edge 1 Move Algorithms: From F/R/U/L/D/B to F [BRE, N]					
To F [BRE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRE	N	TLE	S		
BRE	N	RTE	W		
BRE	N	BRE	N	No Move	0
BRE	N	LBE	E		
To F [BRE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRE	N	TLE	S		
BRE	N	RTE	W		
BRE	N	BRE	N		
BRE	N	LBE	E		
To F [BRE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRE	N	TLE	S		
BRE	N	RTE	W		
BRE	N	BRE	N		
BRE	N	LBE	E		
To F [BRE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRE	N	TLE	S		
BRE	N	RTE	W		
BRE	N	BRE	N		
BRE	N	LBE	E		
To F [BRE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRE	N	TLE	S		
BRE	N	RTE	W		
BRE	N	BRE	N		
BRE	N	LBE	E		
To F [BRE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRE	N	TLE	S		
BRE	N	RTE	W		
BRE	N	BRE	N		
BRE	N	LBE	E		

Step 8 – R4 Edge 1 Move Algorithms: From F/R/U/L/D/B to F [LBE, N]					
To F [LBE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LBE	N	TLE	E		
LBE	N	RTE	S		
LBE	N	BRE	W		
LBE	N	LBE	N	No move	0
To F [LBE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LBE	N	TLE	E		
LBE	N	RTE	S		
LBE	N	BRE	W		
LBE	N	LBE	N		
To F [LBE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LBE	N	TLE	E		
LBE	N	RTE	S		
LBE	N	BRE	W		
LBE	N	LBE	N		
To F [LBE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LBE	N	TLE	E		
LBE	N	RTE	S		
LBE	N	BRE	W		
LBE	N	LBE	N		
To F [LBE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LBE	N	TLE	E		
LBE	N	RTE	S		
LBE	N	BRE	W		
LBE	N	LBE	N		
To F [LBE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LBE	N	TLE	E		
LBE	N	RTE	S		
LBE	N	BRE	W		
LBE	N	LBE	N		

R4 Edge 2 Move Algorithms (Check: undone)

Front face center cubies other than the origin and destination center cubies are left unchanged.

Step 9 – R4 Edge 2 Move Algorithms: From F/R/U/L/D/B to F [TRE, N]					
To F [TRE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRE	N	TRE	N	No move	0
TRE	N	RBE	E		
TRE	N	BLE	S		
TRE	N	LTE	W		
To F [TRE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRE	N	TRE	N		
TRE	N	RBE	E		
TRE	N	BLE	S		
TRE	N	LTE	W		
To F [TRE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRE	N	TRE	N		
TRE	N	RBE	E		
TRE	N	BLE	S		
TRE	N	LTE	W		
To F [TRE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRE	N	TRE	N		
TRE	N	RBE	E		
TRE	N	BLE	S		
TRE	N	LTE	W		
To F [TRE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRE	N	TRE	N		
TRE	N	RBE	E		
TRE	N	BLE	S		
TRE	N	LTE	W		
To F [TRE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRE	N	TRE	N		
TRE	N	RBE	E		
TRE	N	BLE	S		
TRE	N	LTE	W		

Step 10 – R4 Edge 2 Move Algorithms: From F/R/U/L/D/B to F [RBE, N]					
To F [RBE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RBE	N	TRE	W		
RBE	N	RBE	N	No Move	0
RBE	N	BLE	E		
RBE	N	LTE	S		
To F [RBE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RBE	N	TRE	W		
RBE	N	RBE	N		
RBE	N	BLE	E		
RBE	N	LTE	S		
To F [RBE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RBE	N	TRE	W		
RBE	N	RBE	N		
RBE	N	BLE	E		
RBE	N	LTE	S		
To F [RBE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RBE	N	TRE	W		
RBE	N	RBE	N		
RBE	N	BLE	E		
RBE	N	LTE	S		
To F [RBE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RBE	N	TRE	W		
RBE	N	RBE	N		
RBE	N	BLE	E		
RBE	N	LTE	S		
To F [RBE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
RBE	N	TRE	W		
RBE	N	RBE	N		
RBE	N	BLE	E		
RBE	N	LTE	S		

Step 11 – R4 Edge 2 Move Algorithms: From F/R/U/L/D/B to F [BLE, N]					
To F [BLE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLE	N	TRE	S		
BLE	N	RBE	W		
BLE	N	BLE	N	No Move	0
BLE	N	LTE	E		
To F [BLE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLE	N	TRE	S		
BLE	N	RBE	W		
BLE	N	BLE	N		
BLE	N	LTE	E		
To F [BLE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLE	N	TRE	S		
BLE	N	RBE	W		
BLE	N	BLE	N		
BLE	N	LTE	E		
To F [BLE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLE	N	TRE	S		
BLE	N	RBE	W		
BLE	N	BLE	N		
BLE	N	LTE	E		
To F [BLE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLE	N	TRE	S		
BLE	N	RBE	W		
BLE	N	BLE	N		
BLE	N	LTE	E		
To F [BLE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLE	N	TRE	S		
BLE	N	RBE	W		
BLE	N	BLE	N		
BLE	N	LTE	E		

Step 12 – R4 Edge 2 Move Algorithms: From F/R/U/L/D/B to F [LTE, N]					
To F [LTE, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LTE	N	TRE	E		
LTE	N	RBE	S		
LTE	N	BLE	W		
LTE	N	LTE	N	No move	0
To F [LTE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LTE	N	TRE	E		
LTE	N	RBE	S		
LTE	N	BLE	W		
LTE	N	LTE	N		
To F [LTE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LTE	N	TRE	E		
LTE	N	RBE	S		
LTE	N	BLE	W		
LTE	N	LTE	N		
To F [LTE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LTE	N	TRE	E		
LTE	N	RBE	S		
LTE	N	BLE	W		
LTE	N	LTE	N		
To F [LTE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LTE	N	TRE	E		
LTE	N	RBE	S		
LTE	N	BLE	W		
LTE	N	LTE	N		
To F [LTE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
LTE	N	TRE	E		
LTE	N	RBE	S		
LTE	N	BLE	W		
LTE	N	LTE	N		

R4 Corner Move Algorithms (Check: undone)

Front face center cubies other than the origin and destination center cubies are left unchanged.

Step 13 – R4 Corner Move Algorithms: From F/R/U/L/D/B to F [TLC, N]					
To F [TLC, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLC	N	TLC	N	No move	0
TLC	N	TRC	E		
TLC	N	BRC	S		
TLC	N	BLC	W		
To F [TLC, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLC	N	TLC	N		
TLC	N	TRC	E		
TLC	N	BRC	S		
TLC	N	BLC	W		
To F [TLC, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLC	N	TLC	N		
TLC	N	TRC	E		
TLC	N	BRC	S		
TLC	N	BLC	W		
To F [TLC, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLC	N	TLC	N		
TLC	N	TRC	E		
TLC	N	BRC	S		
TLC	N	BLC	W		
To F [TLC, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLC	N	TLC	N		
TLC	N	TRC	E		
TLC	N	BRC	S		
TLC	N	BLC	W		
To F [TLC, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TLC	N	TLC	N		
TLC	N	TRC	E		
TLC	N	BRC	S		
TLC	N	BLC	W		

Step 14 – R4 Corner Move Algorithms: From F/R/U/L/D/B to F [TRC, N]					
To F [TRC, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRC	N	TLC	W		
TRC	N	TRC	N	No Move	0
TRC	N	BRC	E		
TRC	N	BLC	S		
To F [RBE, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRC	N	TLC	W		
TRC	N	TRC	N		
TRC	N	BRC	E		
TRC	N	BLC	S		
To F [RBE, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRC	N	TLC	W		
TRC	N	TRC	N		
TRC	N	BRC	E		
TRC	N	BLC	S		
To F [RBE, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRC	N	TLC	W		
TRC	N	TRC	N		
TRC	N	BRC	E		
TRC	N	BLC	S		
To F [RBE, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRC	N	TLC	W		
TRC	N	TRC	N		
TRC	N	BRC	E		
TRC	N	BLC	S		
To F [RBE, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
TRC	N	TLC	W		
TRC	N	TRC	N		
TRC	N	BRC	E		
TRC	N	BLC	S		

Step 15 – R4 Corner Move Algorithms: From F/R/U/L/D/B to F [BRC, N]					
To F [BRC, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRC	N	TLC	S		
BRC	N	TRC	W		
BRC	N	BRC	N	No Move	0
BRC	N	BLC	E		
To F [BRC, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRC	N	TLC	S		
BRC	N	TRC	W		
BRC	N	BRC	N		
BRC	N	BLC	E		
To F [BRC, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRC	N	TLC	S		
BRC	N	TRC	W		
BRC	N	BRC	N		
BRC	N	BLC	E		
To F [BRC, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRC	N	TLC	S		
BRC	N	TRC	W		
BRC	N	BRC	N		
BRC	N	BLC	E		
To F [BRC, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRC	N	TLC	S		
BRC	N	TRC	W		
BRC	N	BRC	N		
BRC	N	BLC	E		
To F [BRC, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BRC	N	TLC	S		
BRC	N	TRC	W		
BRC	N	BRC	N		
BRC	N	BLC	E		

Step 16 – R4 Corner Move Algorithms: From F/R/U/L/D/B to F [BLC, N]					
To F [BLC, N]		**From F**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLC	N	TLC	E		
BLC	N	TRC	S		
BLC	N	BRC	W		
BLC	N	BLC	N	No move	0
To F [BLC, N]		**From R**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLC	N	TLC	E		
BLC	N	TRC	S		
BLC	N	BRC	W		
BLC	N	BLC	N		
To F [BLC, N]		**From U**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLC	N	TLC	E		
BLC	N	TRC	S		
BLC	N	BRC	W		
BLC	N	BLC	N		
To F [BLC, N]		**From L**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLC	N	TLC	E		
BLC	N	TRC	S		
BLC	N	BRC	W		
BLC	N	BLC	N		
To F [BLC, N]		**From D**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLC	N	TLC	E		
BLC	N	TRC	S		
BLC	N	BRC	W		
BLC	N	BLC	N		
To F [BLC, N]		**From B**			
Location	Orientation	Location	Orientation	Algorithms	Moves
BLC	N	TLC	E		
BLC	N	TRC	S		
BLC	N	BRC	W		
BLC	N	BLC	N		

4x4x4 Cube Wire Grid Model

Using a wire grid model, it is easy to see through the cube where letters are. This model can be used with pencil and rubber to find a path on the cube.

This model may be used to find algorithms for moving a single character from a location to another.