

D.K.T.E. Society's Textile and Engineering Institute, Ichalkaranji.

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

Accredited with 'A+' Grade by NAAC

Department of Computer Science & Engineering

2020-2021



The project report on

Pocket Cube

[ACM-ICPC]

Under The Guidance Of

Mr. S. J. Murchite

DEVELOPED BY:

- | | |
|---------------------------|----------|
| 1. Gidwani Manav Krupal | 19UCS038 |
| 2. Jadhav Gaurav Nivas | 19UCS045 |
| 3. Ekal Priyanshu Prakash | 19UCS033 |
| 4. Kale Shubhankar Suhas | 19UCS052 |

**D.K.T.E. Society's Textile and Engineering Institute,
Ichalkaranji.**

An Autonomous Institute with NAAC A+ grade

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that

Gidwani Manav Krupal	19UCS038
Jadhav Gaurav Nivas	19UCS045
Ekal Priyanshu Prakash	19UCS033
Kale Shubhankar Suhas	19UCS052

Have successfully completed the project work, entitled,

Pocket Cube

In partial fulfillment for the award of degree of Bachelor of Technology in Computer Science and Engineering. This is the record of their work carried out during academic year 2020-2021.

Date:

Place: Ichalkaranji

Mr. S. J. Murchite

Prof. Dr. D. V. Kodavade

[Project Guide]_____ [External Examiner]_____ [Head of Department]

Prof. Dr. P. V. Kadole

[Director]

DECLARATION

We the undersigned students of S.Y. C.S.E. declare that the Project work report entitled “Pocket Cube” written and submitted under the guidance of Mr. S. J. Murchite is our original work. The empirical findings in this report are based on the data collected by us. The matter assimilated in this report is not reproduction from any readymade report.

Date:

Place: Ichalkaranji

Name	Signature
Gidwani Manav Krupal	19UCS038
Jadhav Gaurav Nivas	19UCS045
Ekal Priyanshu Prakash	19UCS033
Kale Shubhankar Suhas	19UCS052

INDEX

CONTENTS	PAGE
1. Abstract	5
2. Problem Statement	6
3. Problem Description	6
4. Requirement Specification	7
5. Requirement Analysis	8
6. Problem Solution	9
7. Flowcharts	
8. Snap shot	14
9. Conclusion	16
10. Reference	17

ABSTRACT

The mission of the Pocket Cube Project is to count the number of completed faces initially and after each rotation, either clockwise or counter clockwise. Our main goal is to develop this application to show that how a Rubik's 2x2x2 pocket cube is get solved by denoting the output as complete faces.

PROBLEM STATEMENT

To show the maximum number of completed faces in no more than N twist steps of a $2 \times 2 \times 2$ Pocket Cube. Each small face of the cube (indexed from 0-23) forms a color which is denoted by a number. Such 6 numbers (each occurring 4 times) are present on the cube (indexed from 0 to 5).

PROBLEM DESCRIPTION

Pocket Cube is a 3-D combination puzzle. It is a $2 \times 2 \times 2$ cube, that is, it consists of 8 mini cubes. For a combination of 4 cubes (2×2 mini-cubes), sharing a whole cube face, you can twist it 90 degrees in clockwise or counter-clockwise direction. This twist operation is called a one twist step.

Considering all faces of mini-cubes, there are total 24 faces having 6 different colors (indexed from 0 to 5). If any 4 mini-cubes' faces are of same color relying on same large face of cube, we can call that face as completed face.



Now, given you a colour arrangement of all 24 faces from a scrambled Pocket Cube, tell the maximum possible number of completed faces in no more than N twist steps.

Index of each face is shown as below:

		0	1		
		2	3		
4	5	6	7	8	9
10	11	12	13	14	15
		16	17		
		18	19		
		20	21		
		22	23		

Input:

There will be several test cases. In each test case, there will be 2 lines. One integer N ($1 \leq N \leq 7$) in the first line, then 24 integers C_i separated by a single space in the second line. For index $0 \leq i < 24$, C_i is colour of the corresponding face.

Output:

For each test case, output the maximum number of completed faces during no more than N twist steps.

Sample input:

1
0 0 0 0 1 1 2 2 3 3 1 1 2 2 3 3 4 4 4 4 5 5 5 5
1
0 4 0 4 1 1 2 5 3 3 1 1 2 5 3 3 4 0 4 0 5 2 5 2

Sample Output:

6
2

REQUIREMENT SPECIFICATION

1. The input must be an integer valued choice that which position is to specified.
2. There should be total 24 indexes (indexed from 0 to 23) having 6 different integers (from 0 to 5)
3. There must be no more than N twist steps ($1 \leq N \leq 7$).
4. The output must be single integer value that must denote number of complete faces.

REQUIREMENT ANALYSIS

The cube is 2x2x2 dimension. It consists of 8 mini cubes of which a combination of 4 mini cubes sharing a whole face of cube is to be twisted either clockwise or counter clockwise. Considering all the mini-cubes, there are total 24 mini faces having 6 different colors indexed from 0 to 5. These 24 faces are indexed from 0 to 23. Diagrammatically, this indexing is shown as below:

		0 0	1 0		
		2 0	3 0		
4 1	5 1	6 2	7 2	8 3	9 3
10 1	11 1	12 2	13 2	14 3	15 3
		16 4	17 4		
		18 4	19 4		
		20 5	21 5		
		22 5	23 5		

REQUIREMENT ANALYSIS

Here, for **Case 1**, the index 0,1,2,3 has 0,0,0,0 which is a complete face. Similarly, 4,5,10,11 has 1,1,1,1 ; 6,7,12,13 has 2,2,2,2 ; 8,9,14,15 has 3,3,3,3 ; 16,17,18,19 has 4,4,4,4 ; 20,21,22,23 has 5,5,5,5, which are the complete faces.

Index : 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Values: 0 0 0 0 1 1 2 2 3 3 1 1 2 2 3 3 4 4 4 4 5 5 5 5

Also, for **Case 2**, the index 4,5,10,11 having 1,1,1,1 & 8,9,14,15 having 3,3,3,3 is complete faces while remaining are flipped vertically top ward as shown below:

Index : 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Values: 0 4 0 4 1 1 2 5 3 3 1 1 2 5 3 3 4 0 4 0 5 2 5 2

For **Case 3**, the index 6,7,12,13 having 2,2,2,2 & 20,21,22,23 having 5,5,5,5 is complete faces while remaining are flipped horizontally in anticlockwise direction as shown below:

Index : 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Values: 4 4 0 0 3 1 2 2 3 1 3 1 2 2 3 1 4 4 0 0 5 5 5 5

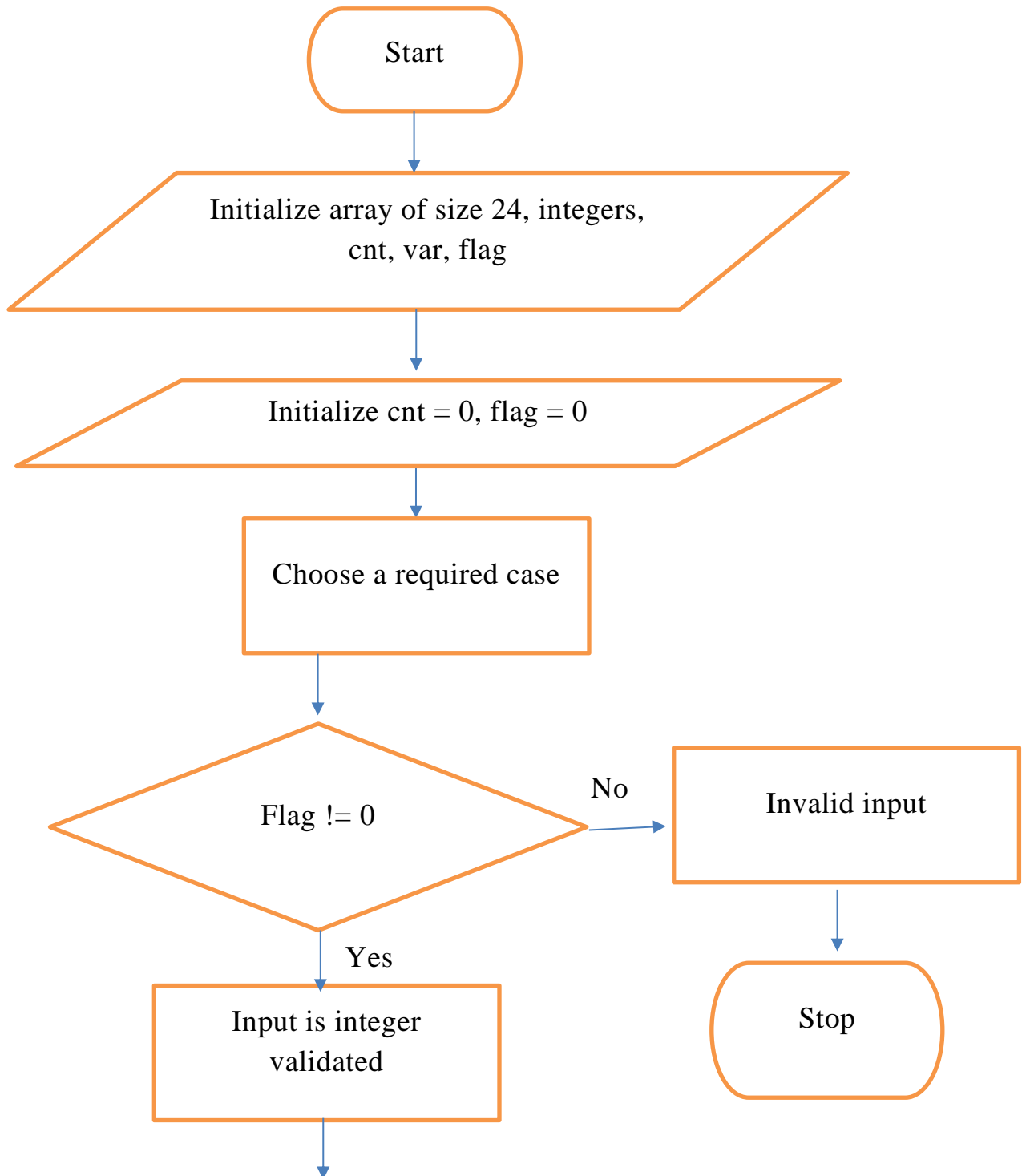
For **Case 4**, the index 6,7,12,13 having 2,2,2,2 & 20,21,22,23 having 5,5,5,5 is complete faces while remaining are flipped horizontally in clockwise direction as shown below:

Index : 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Values: 0 0 4 4 1 3 2 2 1 3 1 3 2 2 1 3 0 0 4 4 5 5 5 5

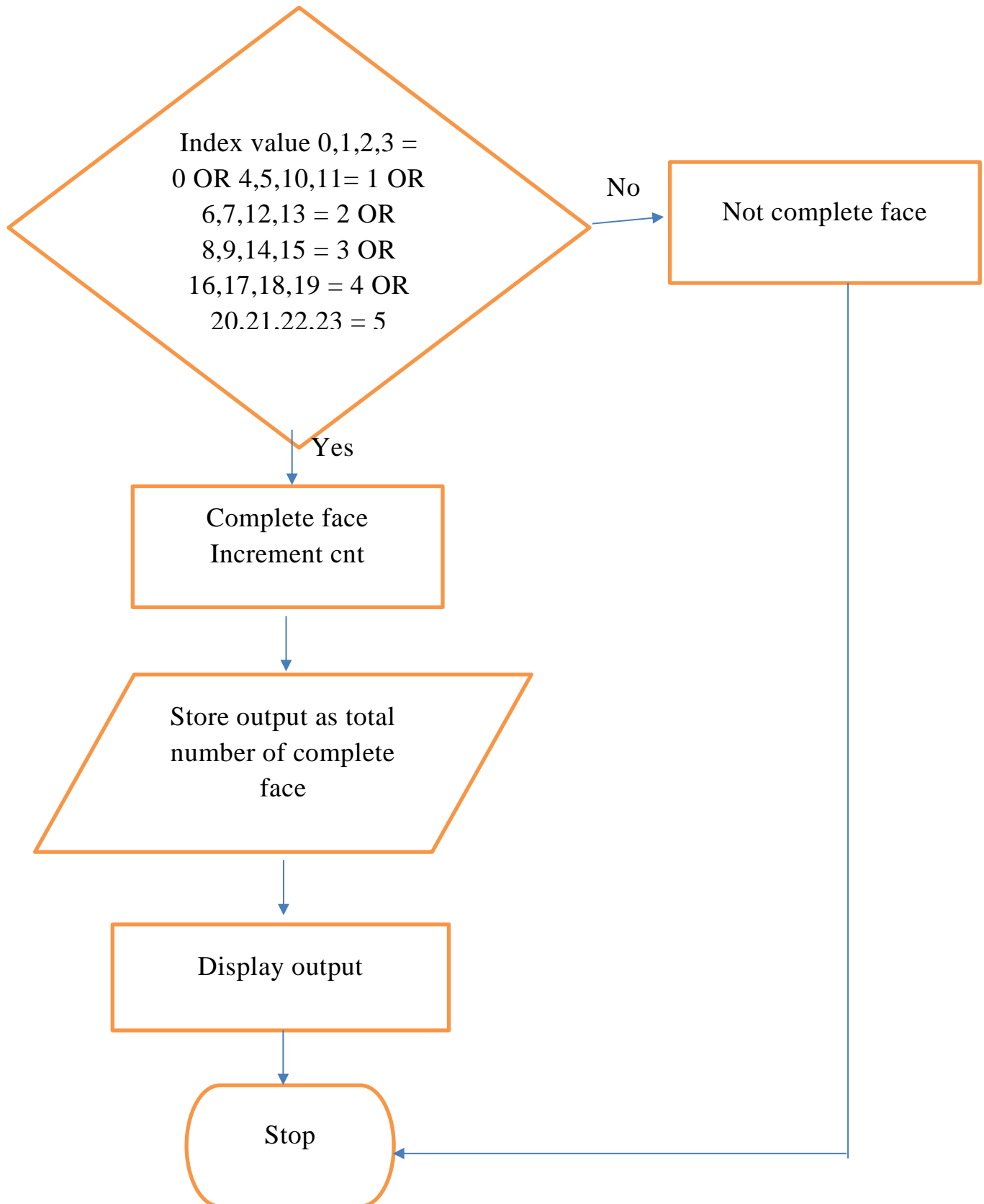
PROBLEM SOLUTION

1. Start
2. Initialize an array C of size 24 as first case, integers i, cnt, var, flag
3. Initialize cnt & flag variable to 0 & 1 respectively.
4. Similar to array C, initialize array B, D, E of size 24 as second, third, fourth case respectively.
5. Ask user to choose the required case.
6. Using for loop initializing i from 0 to length of variable var, if value of var is false then make the flag status equal to 0 & terminate the program.
7. If flag is not equal to 0, then print that input is integer validated.
8. Store the case as integer in var variable.
9. Using switch-case and if-else condition, check out whether the face is complete or not.
10. If complete, increment the cnt variable by 1.
11. Output the variable cnt as number of complete faces.
12. If user inputs a different case, pop-up a message as invalid input.
13. End.

FLOWCHART



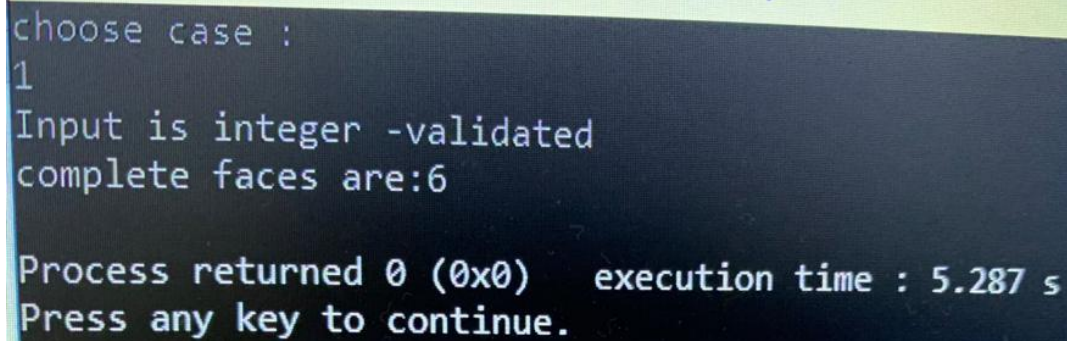
FLOWCHART



SNAP SHOT

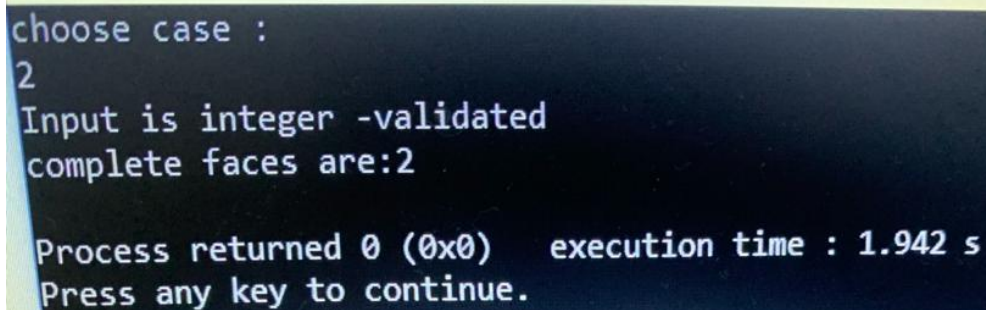
INPUT AND OUTPUTS:

Case 1 :

A terminal window with a dark background and light green text. The text shows a program execution for Case 1. It starts with a prompt 'choose case :', followed by the input '1'. The program then outputs 'Input is integer -validated' and 'complete faces are:6'. Below this, it shows 'Process returned 0 (0x0) execution time : 5.287 s' and 'Press any key to continue.' with a cursor on the next line.

```
choose case :  
1  
Input is integer -validated  
complete faces are:6  
  
Process returned 0 (0x0) execution time : 5.287 s  
Press any key to continue.  
_
```

Case 2 :

A terminal window with a dark background and light green text. The text shows a program execution for Case 2. It starts with a prompt 'choose case :', followed by the input '2'. The program then outputs 'Input is integer -validated' and 'complete faces are:2'. Below this, it shows 'Process returned 0 (0x0) execution time : 1.942 s' and 'Press any key to continue.' with a cursor on the next line.

```
choose case :  
2  
Input is integer -validated  
complete faces are:2  
  
Process returned 0 (0x0) execution time : 1.942 s  
Press any key to continue.  
_
```

Case 3 :

```
choose case :  
3  
Input is integer -validated  
complete faces are:2  
  
Process returned 0 (0x0)   execution time : 8.215 s  
Press any key to continue.
```

Case 4 :

```
choose case :  
4  
Input is integer -validated  
complete faces are:2  
  
Process returned 0 (0x0)   execution time : 2.813 s  
Press any key to continue.
```


Case 5 :

```
choose case :  
a  
Invalid input program aborted  
Process returned 0 (0x0)   execution time : 2.832 s  
Press any key to continue.
```

```
choose case :  
5  
Input is integer -validated  
invalid case  
[  
<Process returned 0 (0x0)   execution time : 3.234 s  
Press any key to continue.
```

CONCLUSION

As an overall conclusion we have achieved the purpose of the project to find out the number of complete faces of the cube based on the given test cases after the N twist steps.

REFERENCES

Books: Let Us C++ by Yashwant Kanetkar

Web-links:

<https://ruwix.com/twisty-puzzles/2x2x2-rubiks-cube-pocket/>