# IF2211 STRATEGI ALGORITMA

LAPORAN TUGAS KECIL 3
Penyelesaian Persoalan 15-Puzzle dengan Algoritma Branch and Bound

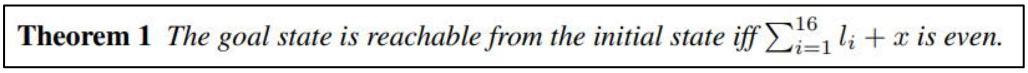
Oleh:

| | |
|---|---|
| Tri Sulton Adila | 13520033 |

**PROGRAM STUDI TEKNIK INFORMATIKA**
**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**
**INSTITUT TEKNOLOGI BANDUNG**
**2022**

## A. Cara Kerja Algoritma *Branch and Bound*

Apabila suatu puzzle merupakan puzzle yang solveable berdasarkan teorema 1, algoritma berikut dapat dilanjutkan untuk memecahkan persoalan 15-Puzzle

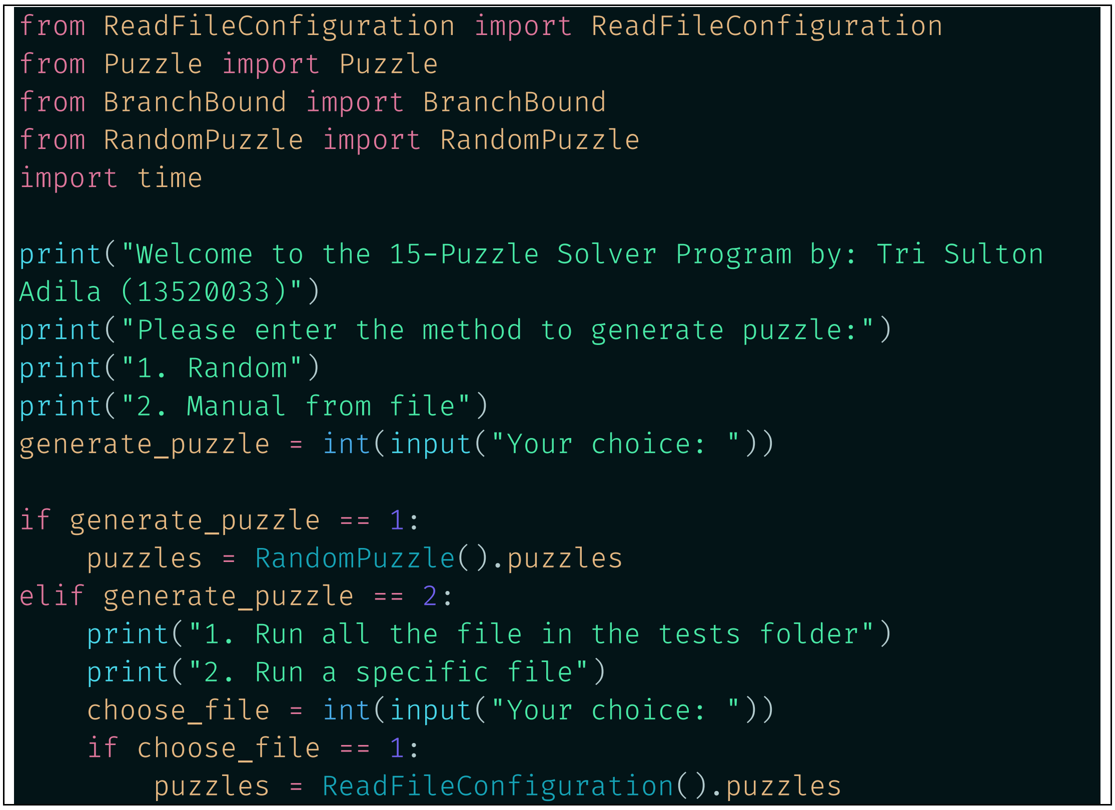


**Theorem 1** *The goal state is reachable from the initial state iff* $\sum_{i=1}^{16} l_i + x$ *is even.*

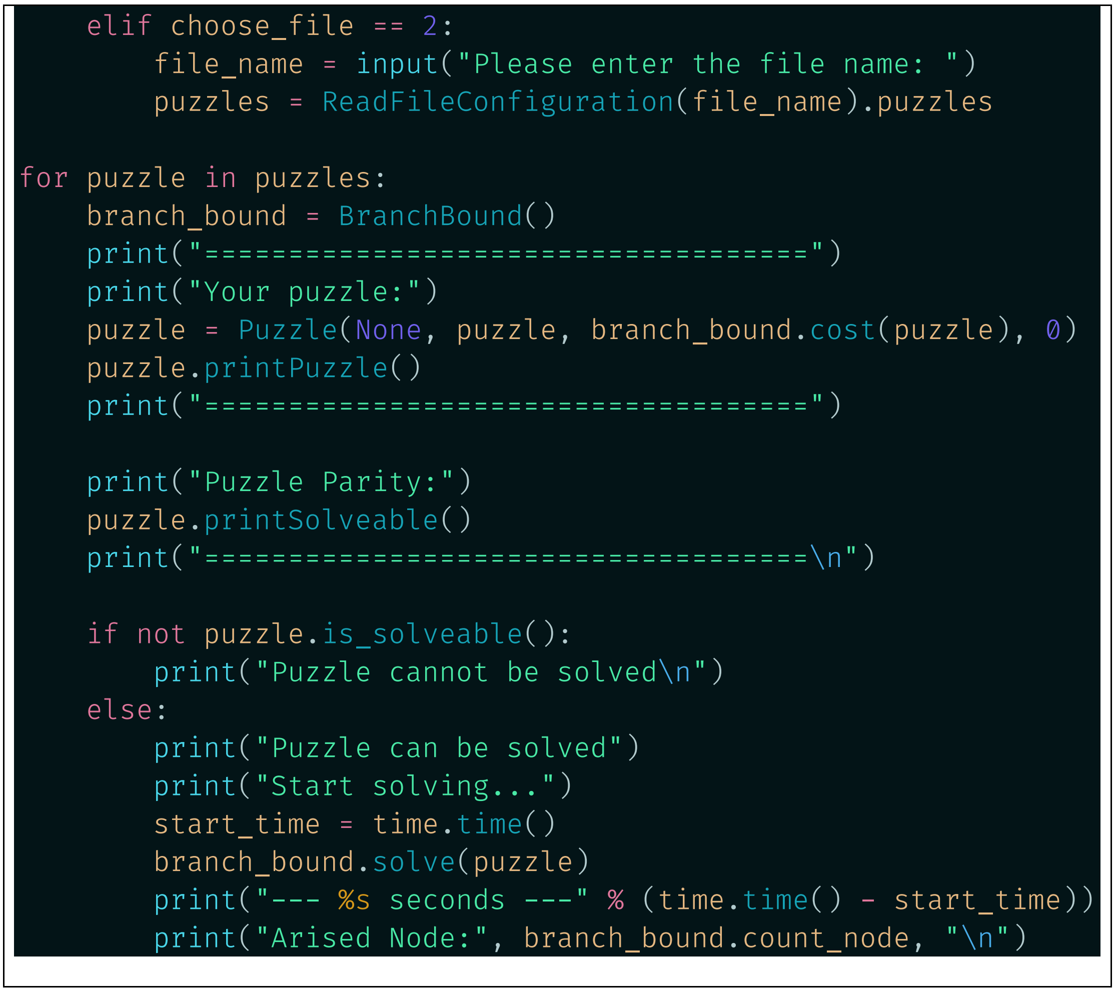
1. Masukkan puzzle yang ingin diselesaikan ke dalam priority queue (cost terkecil diprioritaskan)
2. Ambil puzzle dari queue, lalu, masukkan puzzle tersebut ke dalam sebuah list “telah dikunjungi”
3. Apabila puzzle yang diambil dari queue merupakan solusi, hentikan pencarian
4. Jika tidak, bangkitkan semua kemungkinan puzzle baru berdasarkan arah gerak slot kosong
5. Hitung cost puzzle baru berdasarkan banyaknya langkah dari root node ke puzzle ditambah dengan banyaknya nomor selain slot kosong yang tidak berada di tempatnya berdasarkan solusi
6. Periksa setiap puzzle baru apakah telah berada di dalam list “telah dikunjungi” atau tidak
7. Apabila tidak, masukkan puzzle baru ke dalam queue
8. Apabila iya, tidak perlu memasukkannya ke dalam queue
9. Ulangi mulai dari langkah 2 sampai solusi ditemukan atau isi dari queue habis (berdasarkan asumsi awal bahwa puzzle pasti solveable, isi queue sebenarnya tidak akan pernah habis)

## B. Source Code Program

1. main.py

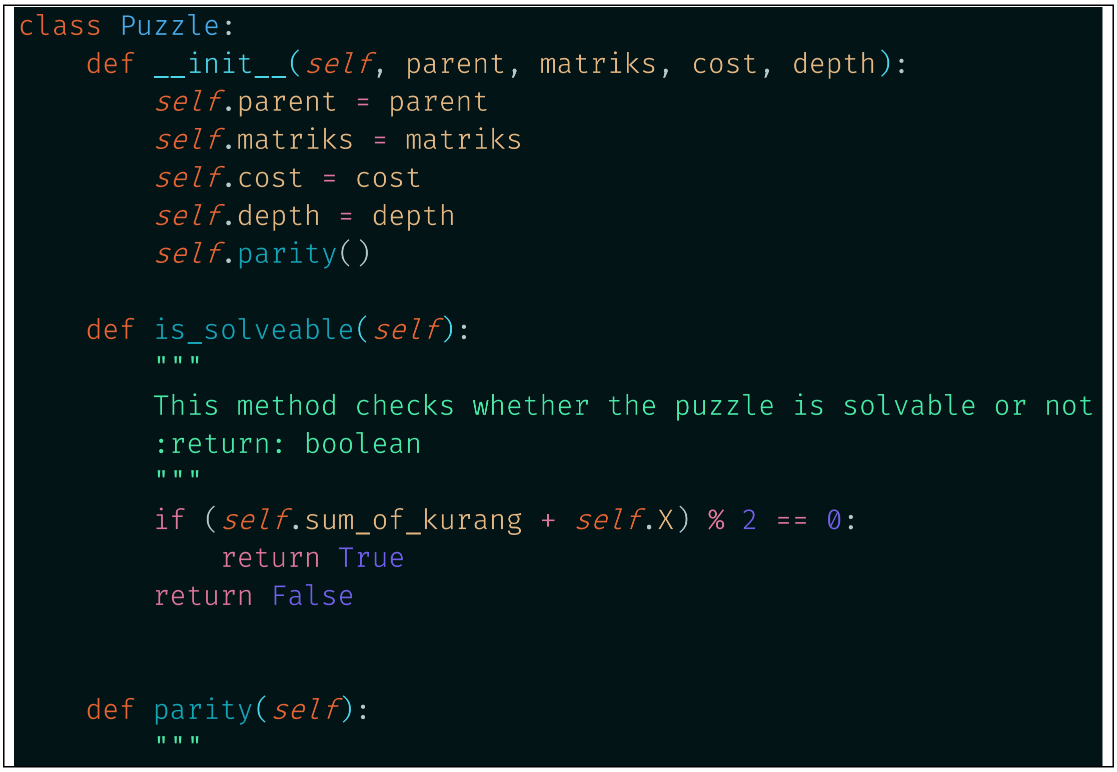


```python
from ReadFileConfiguration import ReadFileConfiguration
from Puzzle import Puzzle
from BranchBound import BranchBound
from RandomPuzzle import RandomPuzzle
import time

print("Welcome to the 15-Puzzle Solver Program by: Tri Sulton
Adila (13520033)")
print("Please enter the method to generate puzzle:")
print("1. Random")
print("2. Manual from file")
generate_puzzle = int(input("Your choice: "))

if generate_puzzle == 1:
    puzzles = RandomPuzzle().puzzles
elif generate_puzzle == 2:
    print("1. Run all the file in the tests folder")
    print("2. Run a specific file")
    choose_file = int(input("Your choice: "))
    if choose_file == 1:
        puzzles = ReadFileConfiguration().puzzles
```

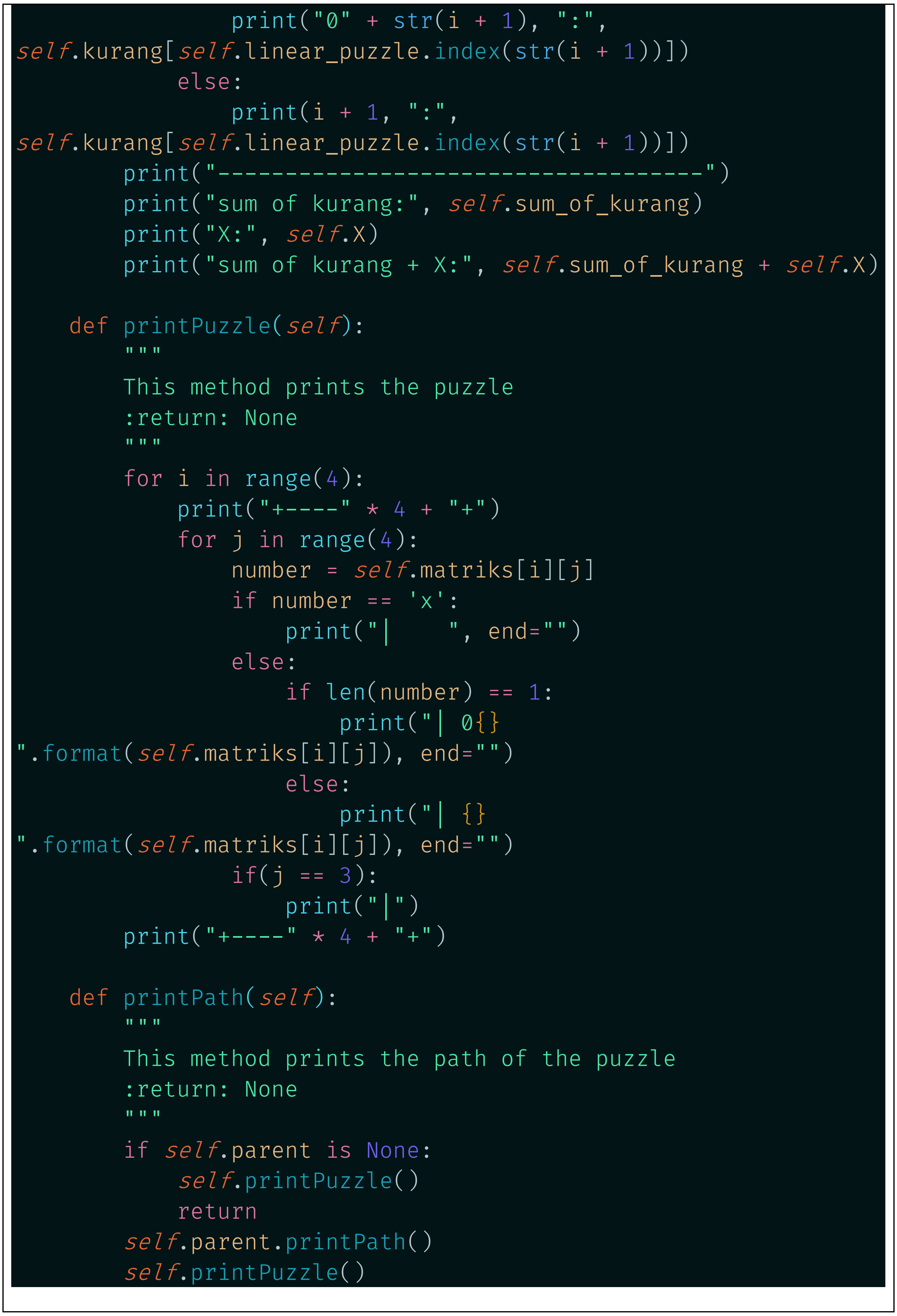


```
    elif choose_file == 2:
        file_name = input("Please enter the file name: ")
        puzzles = ReadFileConfiguration(file_name).puzzles

for puzzle in puzzles:
    branch_bound = BranchBound()
    print("====================================")
    print("Your puzzle:")
    puzzle = Puzzle(None, puzzle, branch_bound.cost(puzzle), 0)
    puzzle.printPuzzle()
    print("====================================")

    print("Puzzle Parity:")
    puzzle.printSolveable()
    print("====================================\n")

    if not puzzle.is_solveable():
        print("Puzzle cannot be solved\n")
    else:
        print("Puzzle can be solved")
        print("Start solving...")
        start_time = time.time()
        branch_bound.solve(puzzle)
        print("--- %s seconds ---" % (time.time() - start_time))
        print("Arised Node:", branch_bound.count_node, "\n")
```

2. Puzzle.py



```
class Puzzle:
    def __init__(self, parent, matriks, cost, depth):
        self.parent = parent
        self.matriks = matriks
        self.cost = cost
        self.depth = depth
        self.parity()

    def is_solveable(self):
        """
        This method checks whether the puzzle is solvable or not
        :return: boolean
        """
        if (self.sum_of_kurang + self.X) % 2 == 0:
            return True
        return False


    def parity(self):
        """
        This method calculates the parity of the puzzle
        """
        self.linear_puzzle = [number for row in self.matriks for
number in row]
        idx_empty = self.linear_puzzle.index('x')
        self.linear_puzzle[idx_empty] = '16'
        self.kurang = []
        for i in range(16):
            count = 0
            for j in range(i + 1, 16):
                if int(self.linear_puzzle[i]) >
int(self.linear_puzzle[j]):
                    count += 1
            self.kurang.append(count)

        x_row = self.get_x_row()
        x_col = self.get_x_col()
        x_spot = x_row + x_col
        self.sum_of_kurang = 0
        for i in range(16):
            self.sum_of_kurang += int(self.kurang[i])
        if x_spot % 2 == 0:
            self.X = 0
        else:
            self.X = 1

    def get_x_row(self) -> int:
        """
        This method returns the x position of the empty space
        :return: list of int
        """
        for(row, col) in enumerate(self.matriks):
            if 'x' in col:
                return row

    def get_x_col(self) -> int:
        """
        This method returns the x position of the empty space
        :return: list of int
        """
        row = self.get_x_row()
        for(col, number) in enumerate(self.matriks[row]):
            if number == 'x':
                return col

    def printSolveable(self):
        for i in range(16):
            if len(str(i + 1)) == 1:
```



```
                print("0" + str(i + 1), ":",
self.kurang[self.linear_puzzle.index(str(i + 1))])
            else:
                print(i + 1, ":",
self.kurang[self.linear_puzzle.index(str(i + 1))])
        print("------------------------------------")
        print("sum of kurang:", self.sum_of_kurang)
        print("X:", self.X)
        print("sum of kurang + X:", self.sum_of_kurang + self.X)

    def printPuzzle(self):
        """
        This method prints the puzzle
        :return: None
        """
        for i in range(4):
            print("+----" * 4 + "+")
            for j in range(4):
                number = self.matriks[i][j]
                if number == 'x':
                    print("|    ", end="")
                else:
                    if len(number) == 1:
                        print("| 0{}
".format(self.matriks[i][j]), end="")
                    else:
                        print("| {}
".format(self.matriks[i][j]), end="")
                if(j == 3):
                    print("|")
        print("+----" * 4 + "+")

    def printPath(self):
        """
        This method prints the path of the puzzle
        :return: None
        """
        if self.parent is None:
            self.printPuzzle()
            return
        self.parent.printPath()
        self.printPuzzle()
```

3. BranchBound.py

```
from PriorityQueue import PriorityQueue
from Puzzle import Puzzle
import copy
class BranchBound:
    def __init__(self):
        self.queue = PriorityQueue()
        self.goalPuzzle = [['1', '2', '3', '4'],
                           ['5', '6', '7', '8'],
                           ['9', '10', '11', '12'],
                           ['13', '14', '15', 'x']]
        self.visited = []

    def cost(self, matriks) -> int:
        """
        calculate cost of matriks
        """
        cost =0
        for i in range(4):
            for j in range(4):
                tile = matriks[i][j]
                if tile != 'x' and tile != self.goalPuzzle[i][j]:
                    cost += 1
        return cost

    def create_children(self, puzzle) -> list:
        """
        This method creates children of the puzzle
        """
        children = []
        # bottom, left, top, right
        row = [ 1, 0, -1, 0 ]
        col = [ 0, -1, 0, 1 ]
        for i in range(4):
            x_row = puzzle.get_x_row()
            x_col = puzzle.get_x_col()
            new_row = x_row + row[i]
            new_col = x_col + col[i]
            if 0 <= new_row <= 3 and 0 <= new_col <= 3:
                child_matriks = copy.deepcopy(puzzle.matriks)
                child_matriks[x_row][x_col], child_matriks[new_row][new_col] = child_matriks[new_row][new_col], child_matriks[x_row][x_col]
                if child_matriks not in self.visited:
                    child = Puzzle(puzzle, child_matriks, self.cost(child_matriks) + puzzle.depth, puzzle.depth + 1)
                    children.append(child)
        return children

    def solve(self, puzzleParent):
        """
        This method solves the puzzle
        """
        self.puzzleParent = puzzleParent
        self.queue.push(puzzleParent)
        self.count_node = 1
        while (not self.queue.is_empty()):
            puzzle = self.queue.pop()
            self.visited.append(puzzle.matriks)
            if puzzle.matriks == self.goalPuzzle:
                print("\n====================================\nSolution found")
                puzzle.printPath()
                print("====================================")
                return
            else:
                children = self.create_children(puzzle)
                for child in children:
                        self.queue.push(child)
                        self.count_node += 1
```

4. PriorityQueue.py

```
from Puzzle import Puzzle

class PriorityQueue:
    def __init__(self):
        self.buffer = []

    def push(self, item):
        self.buffer.append(item)
        self.buffer.sort(key=lambda x: x.cost) # sort by cost in ascending order

    def pop(self) -> Puzzle:
        item = self.buffer[0]
        self.buffer.remove(item)
        return item

    def is_empty(self) -> bool:
        return len(self.buffer) == 0

    def printQueue(self):
        for item in self.buffer:
            print(item.cost)
```

5. ReadFileConfiguration.py

```
import os

class ReadFileConfiguration:
    def __init__(self, file_name=None):
        self.file_name = file_name
        self.directory_name = "../tests"
        self.puzzles = self.get_puzzles()

    def get_all_files_in_directory(self):
        """
        This method returns a list of all files in a directory
        :param directory: string
        :return: file_name: list of string
        """
        with os.scandir(self.directory_name) as files:
            return [file.name for file in files if
file.is_file()]

    def get_spesific_file(self, file_name):
        """
        This method returns a spesific puzzle file
        :param file_name: string
        :return: puzzle: list of list of char
        """
        return self.read_puzzle(file_name)

    def read_puzzle(self, file_name):
        """
        This method reads a file and returns the puzzle
        :param file_name: string
        :return: puzzle: list of list of char
        """
        puzzle = []
        with open(self.directory_name + "/" + file_name, 'r') as
f:
            data = f.read()
            for line in data.splitlines():
                # get rid of whitespace
                row = [number for number in line.split()]
                puzzle.append(row)
        return puzzle

    def get_puzzles(self):
        """
        This method returns a list of all puzzle files
        :return: list of puzzle
        """
        if(self.file_name is not None):
            return [self.get_spesific_file(self.file_name)]
        return [self.read_puzzle(file_name) for file_name in 
self.get_all_files_in_directory()]
```

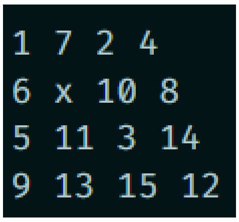
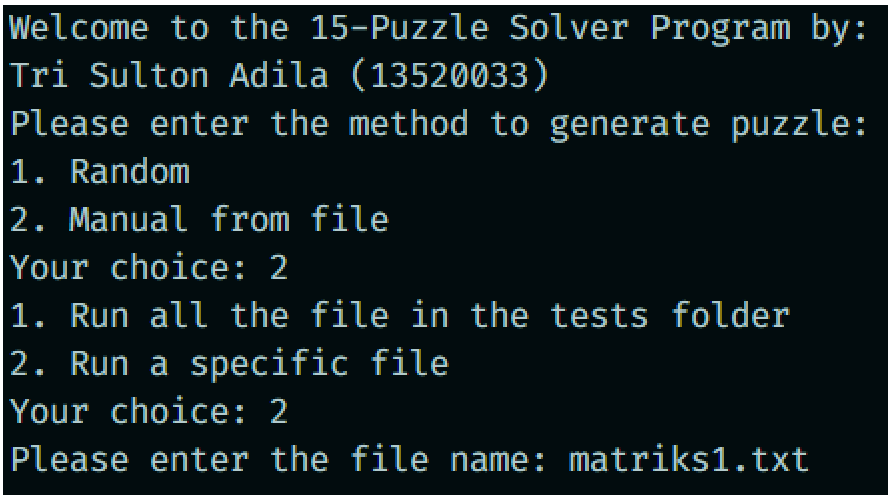
6. RandomPuzzle.py

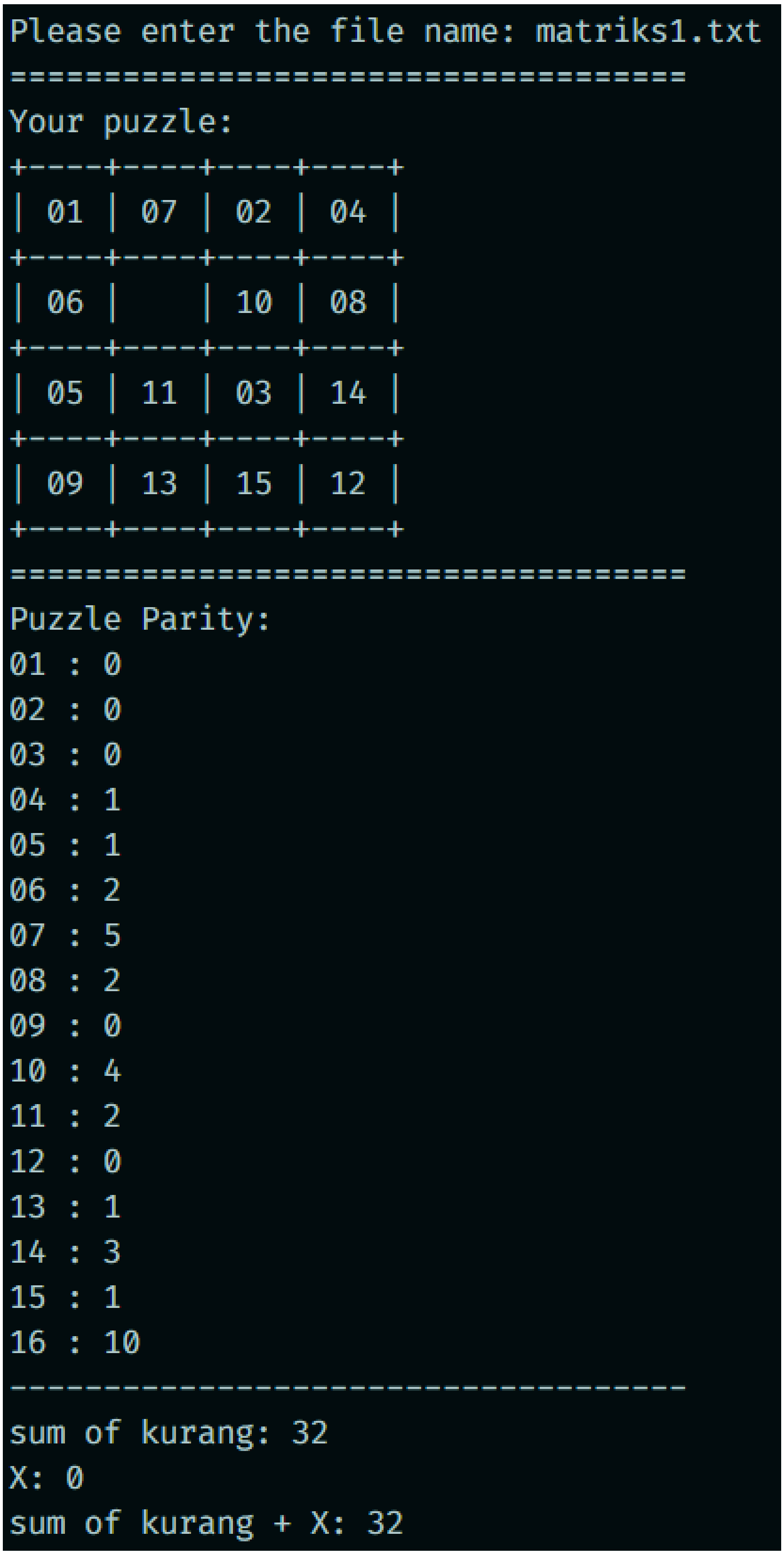
```
import random

class RandomPuzzle:
    number = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '10', 
'11', '12', '13', '14', '15', 'x']
    def __init__(self):
        self.puzzle = []
        self.puzzles = []
        random.shuffle(self.number)
        for i in range(4):
            row = []
            for j in range(4):
                row.append(self.number[i*4 + j])
            self.puzzle.append(row)
        self.puzzles.append(self.puzzle)
```

## C. Screenshoot Hasil Uji

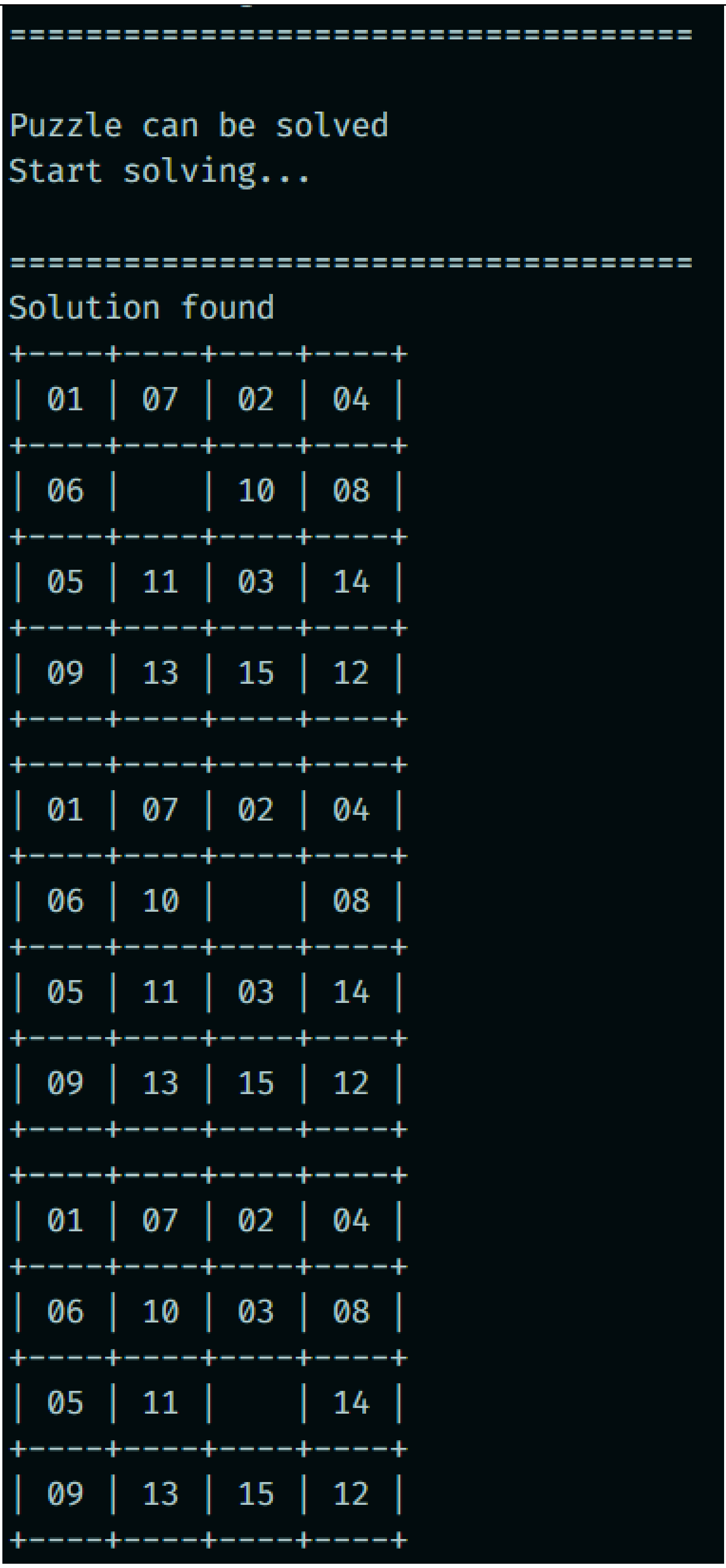
1. Reachable

- matriks1.txt

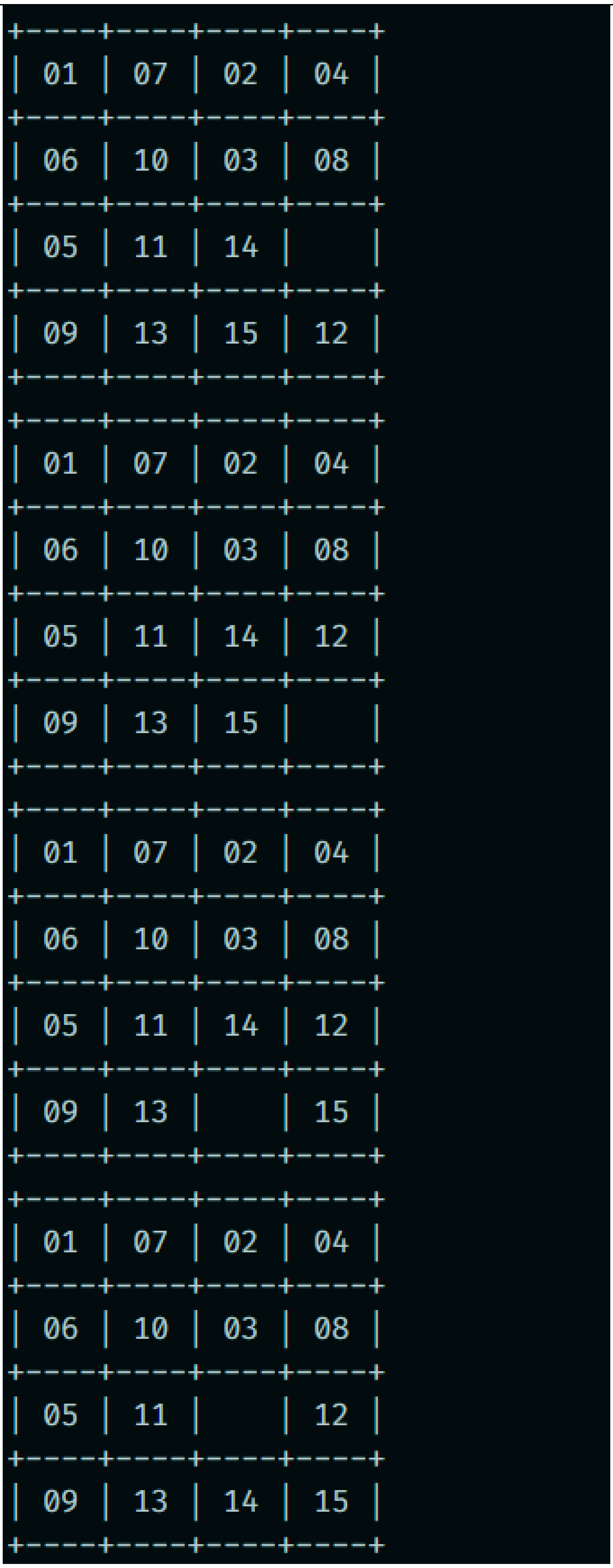
| matriks |
|---|
| <br>1 7 2 4<br>6 x 10 8<br>5 11 3 14<br>9 13 15 12 |
| masukan |
| <br>Welcome to the 15-Puzzle Solver Program by:<br>Tri Sulton Adila (13520033)<br>Please enter the method to generate puzzle:<br>1. Random<br>2. Manual from file<br>Your choice: 2<br>1. Run all the file in the tests folder<br>2. Run a specific file<br>Your choice: 2<br>Please enter the file name: matriks1.txt |
| keluaran |



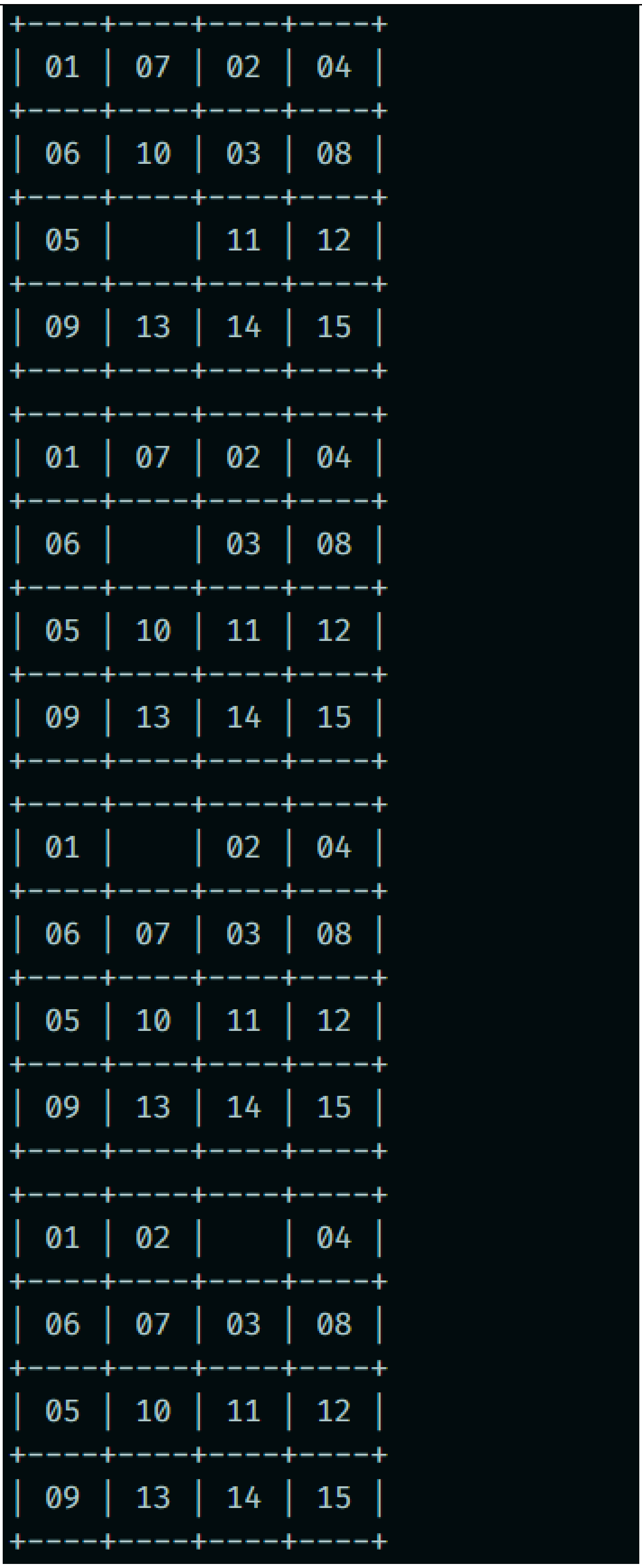
```
Please enter the file name: matriks1.txt
====================================
Your puzzle:
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 |    | 10 | 08 |
+----+----+----+----+
| 05 | 11 | 03 | 14 |
+----+----+----+----+
| 09 | 13 | 15 | 12 |
+----+----+----+----+
====================================
Puzzle Parity:
01 : 0
02 : 0
03 : 0
04 : 1
05 : 1
06 : 2
07 : 5
08 : 2
09 : 0
10 : 4
11 : 2
12 : 0
13 : 1
14 : 3
15 : 1
16 : 10
------------------------------------
sum of kurang: 32
X: 0
sum of kurang + X: 32
```



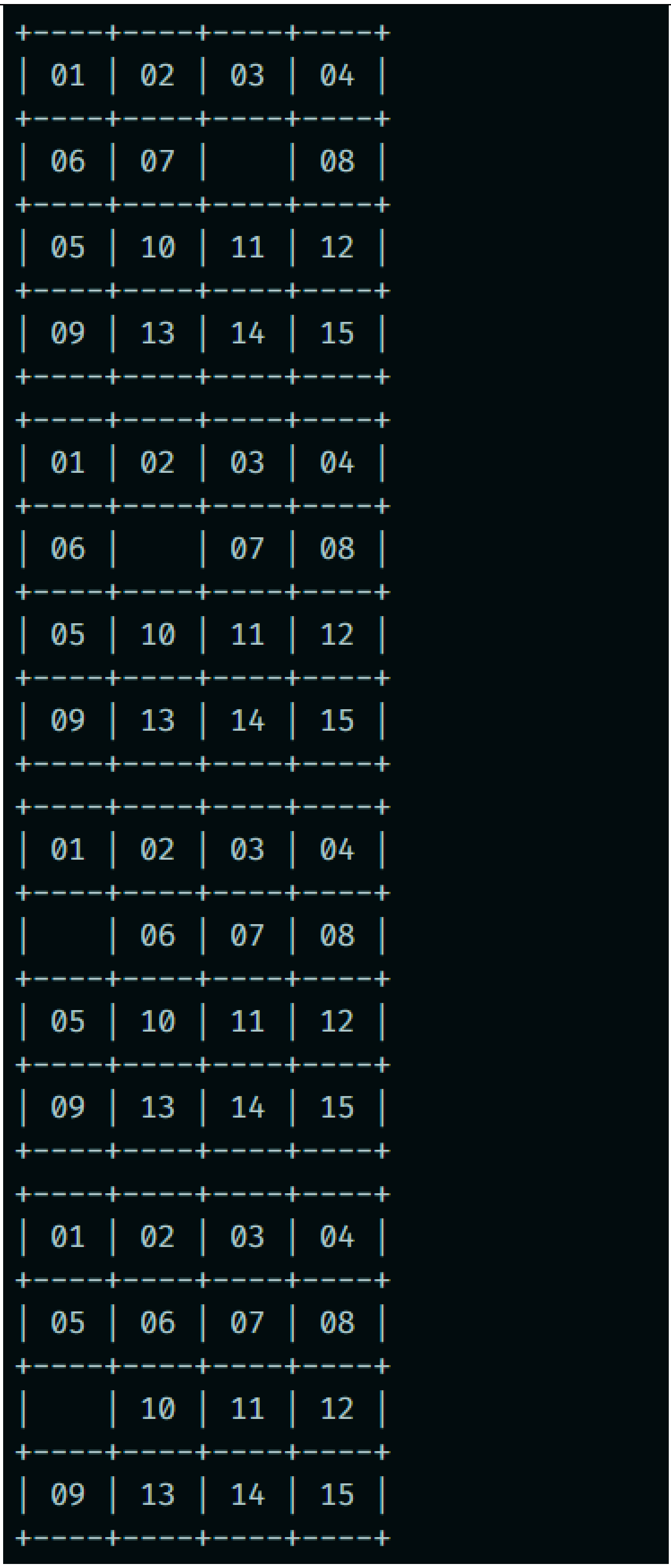
```
========================================

Puzzle can be solved
Start solving...

========================================
Solution found
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 |    | 10 | 08 |
+----+----+----+----+
| 05 | 11 | 03 | 14 |
+----+----+----+----+
| 09 | 13 | 15 | 12 |
+----+----+----+----+
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 |    | 08 |
+----+----+----+----+
| 05 | 11 | 03 | 14 |
+----+----+----+----+
| 09 | 13 | 15 | 12 |
+----+----+----+----+
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 | 03 | 08 |
+----+----+----+----+
| 05 | 11 |    | 14 |
+----+----+----+----+
| 09 | 13 | 15 | 12 |
+----+----+----+----+
```



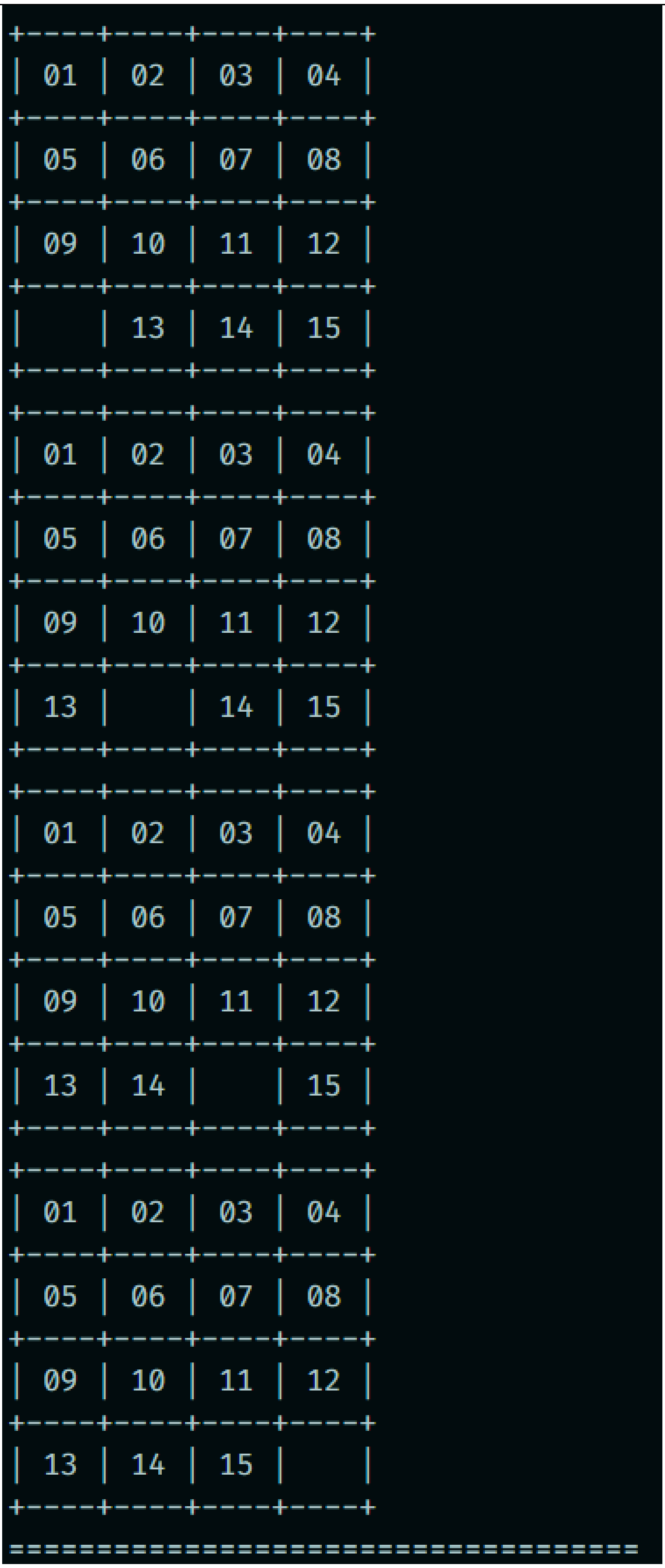
```
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 | 03 | 08 |
+----+----+----+----+
| 05 | 11 | 14 |    |
+----+----+----+----+
| 09 | 13 | 15 | 12 |
+----+----+----+----+
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 | 03 | 08 |
+----+----+----+----+
| 05 | 11 | 14 | 12 |
+----+----+----+----+
| 09 | 13 | 15 |    |
+----+----+----+----+
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 | 03 | 08 |
+----+----+----+----+
| 05 | 11 | 14 | 12 |
+----+----+----+----+
| 09 | 13 |    | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 | 03 | 08 |
+----+----+----+----+
| 05 | 11 |    | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
```



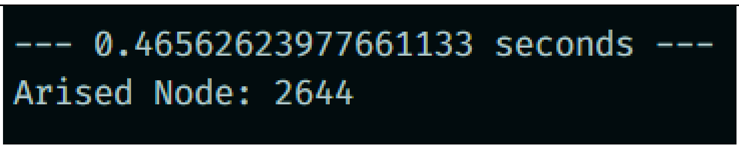
```
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 | 10 | 03 | 08 |
+----+----+----+----+
| 05 |    | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 07 | 02 | 04 |
+----+----+----+----+
| 06 |    | 03 | 08 |
+----+----+----+----+
| 05 | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 |    | 02 | 04 |
+----+----+----+----+
| 06 | 07 | 03 | 08 |
+----+----+----+----+
| 05 | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 |    | 04 |
+----+----+----+----+
| 06 | 07 | 03 | 08 |
+----+----+----+----+
| 05 | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
```



```
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 06 | 07 |    | 08 |
+----+----+----+----+
| 05 | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 06 |    | 07 | 08 |
+----+----+----+----+
| 05 | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
|    | 06 | 07 | 08 |
+----+----+----+----+
| 05 | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
|    | 10 | 11 | 12 |
+----+----+----+----+
| 09 | 13 | 14 | 15 |
+----+----+----+----+
```



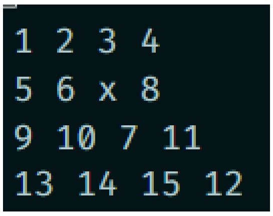
```
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
|    | 13 | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 |    | 14 | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 | 14 |    | 15 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 | 14 | 15 |    |
+----+----+----+----+
====================================
```



```
--- 0.46562623977661133 seconds ---
Arised Node: 2644
```

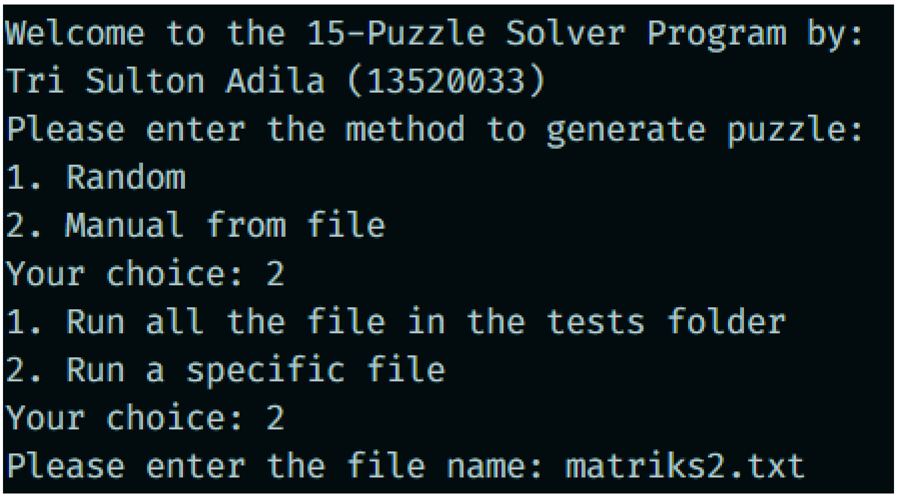
- matriks2.txt

matriks



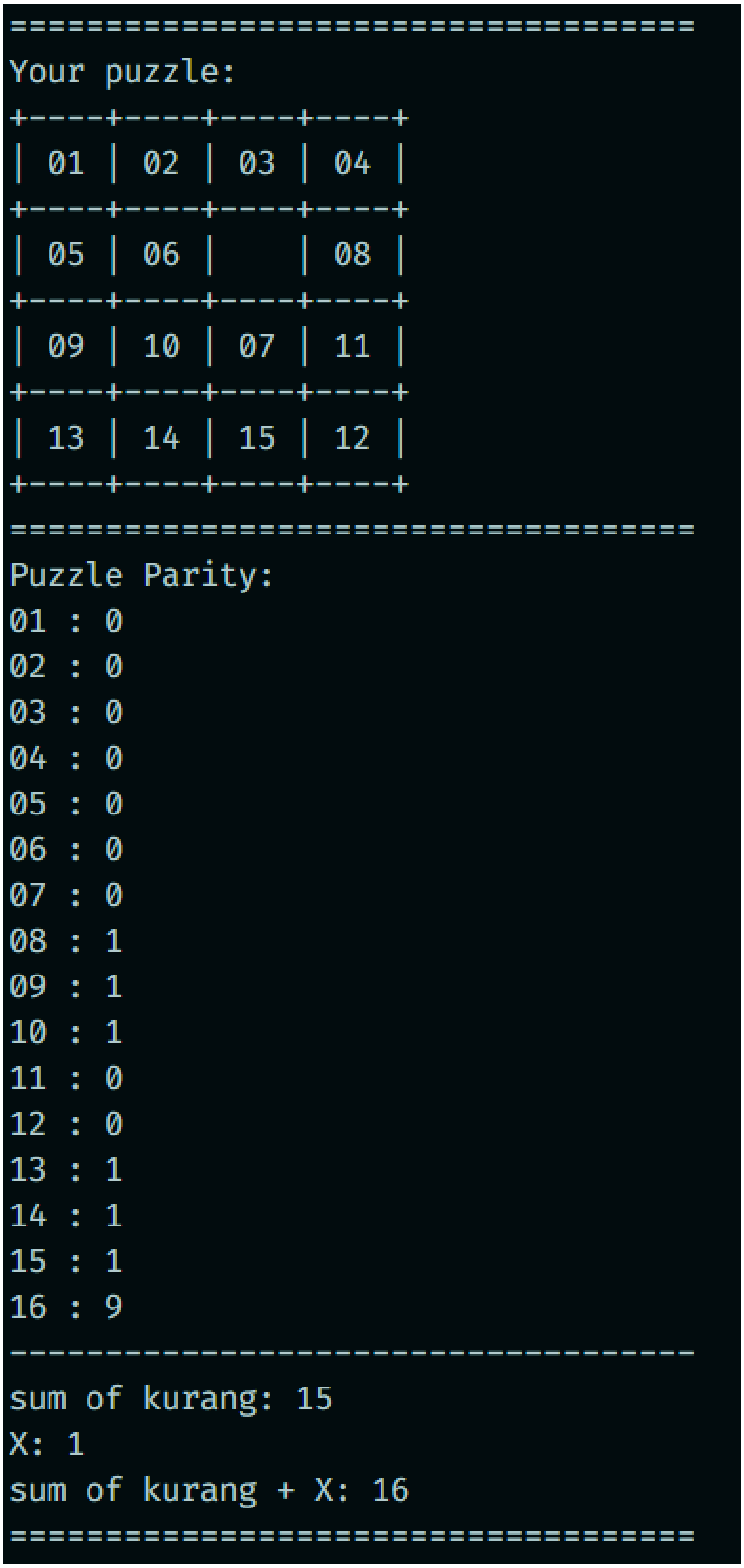
```
1 2 3 4
5 6 x 8
9 10 7 11
13 14 15 12
```

masukan

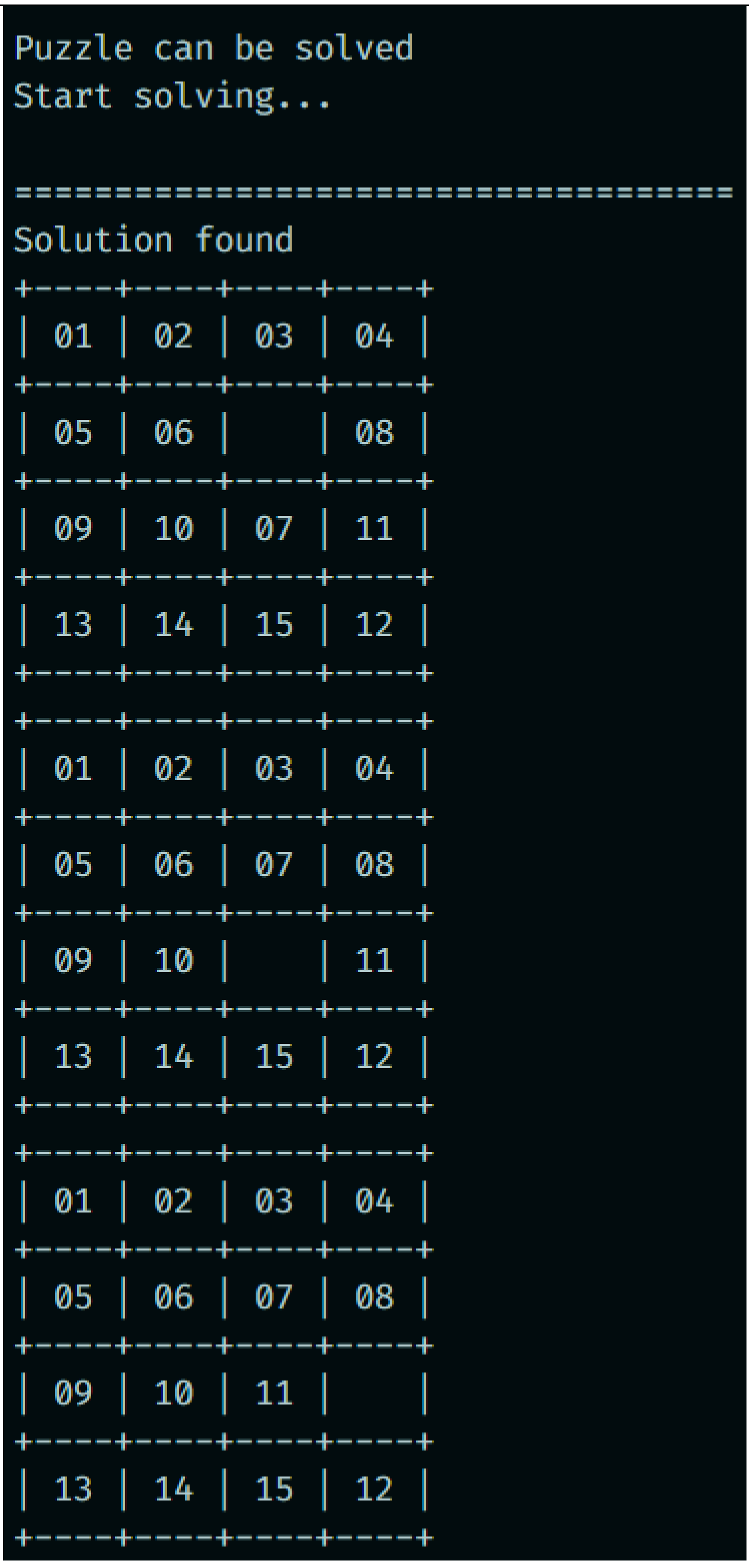


```
Welcome to the 15-Puzzle Solver Program by:
Tri Sulton Adila (13520033)
Please enter the method to generate puzzle:
1. Random
2. Manual from file
Your choice: 2
1. Run all the file in the tests folder
2. Run a specific file
Your choice: 2
Please enter the file name: matriks2.txt
```

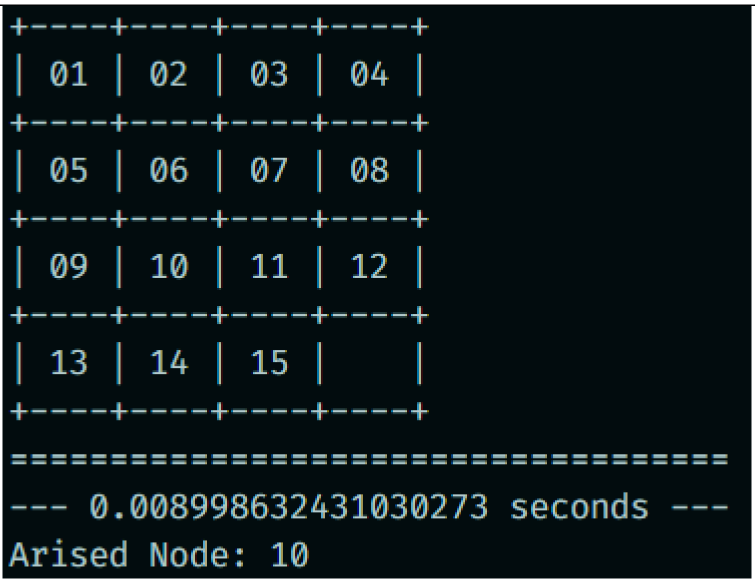
keluaran



```
====================================
Your puzzle:
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 |    | 08 |
+----+----+----+----+
| 09 | 10 | 07 | 11 |
+----+----+----+----+
| 13 | 14 | 15 | 12 |
+----+----+----+----+
====================================
Puzzle Parity:
01 : 0
02 : 0
03 : 0
04 : 0
05 : 0
06 : 0
07 : 0
08 : 1
09 : 1
10 : 1
11 : 0
12 : 0
13 : 1
14 : 1
15 : 1
16 : 9
------------------------------------
sum of kurang: 15
X: 1
sum of kurang + X: 16
====================================
```



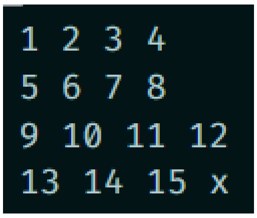
```
Puzzle can be solved
Start solving...

====================================
Solution found
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 |    | 08 |
+----+----+----+----+
| 09 | 10 | 07 | 11 |
+----+----+----+----+
| 13 | 14 | 15 | 12 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 |    | 11 |
+----+----+----+----+
| 13 | 14 | 15 | 12 |
+----+----+----+----+
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 |    |
+----+----+----+----+
| 13 | 14 | 15 | 12 |
+----+----+----+----+
```



```
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 | 14 | 15 |    |
+----+----+----+----+
====================================
--- 0.008998632431030273 seconds ---
Arised Node: 10
```

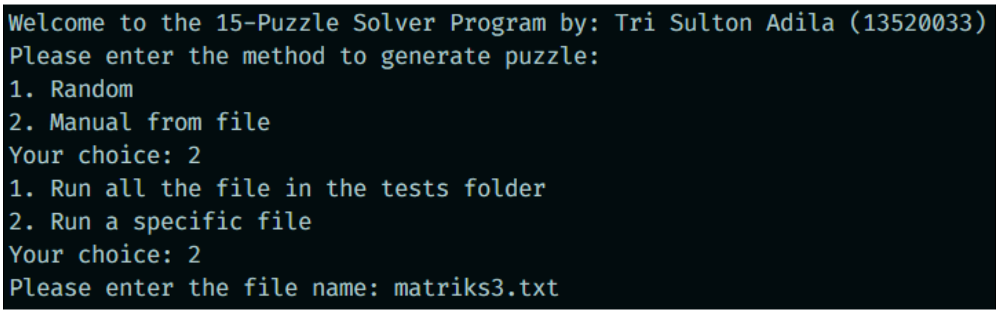
- matriks3.txt

matriks



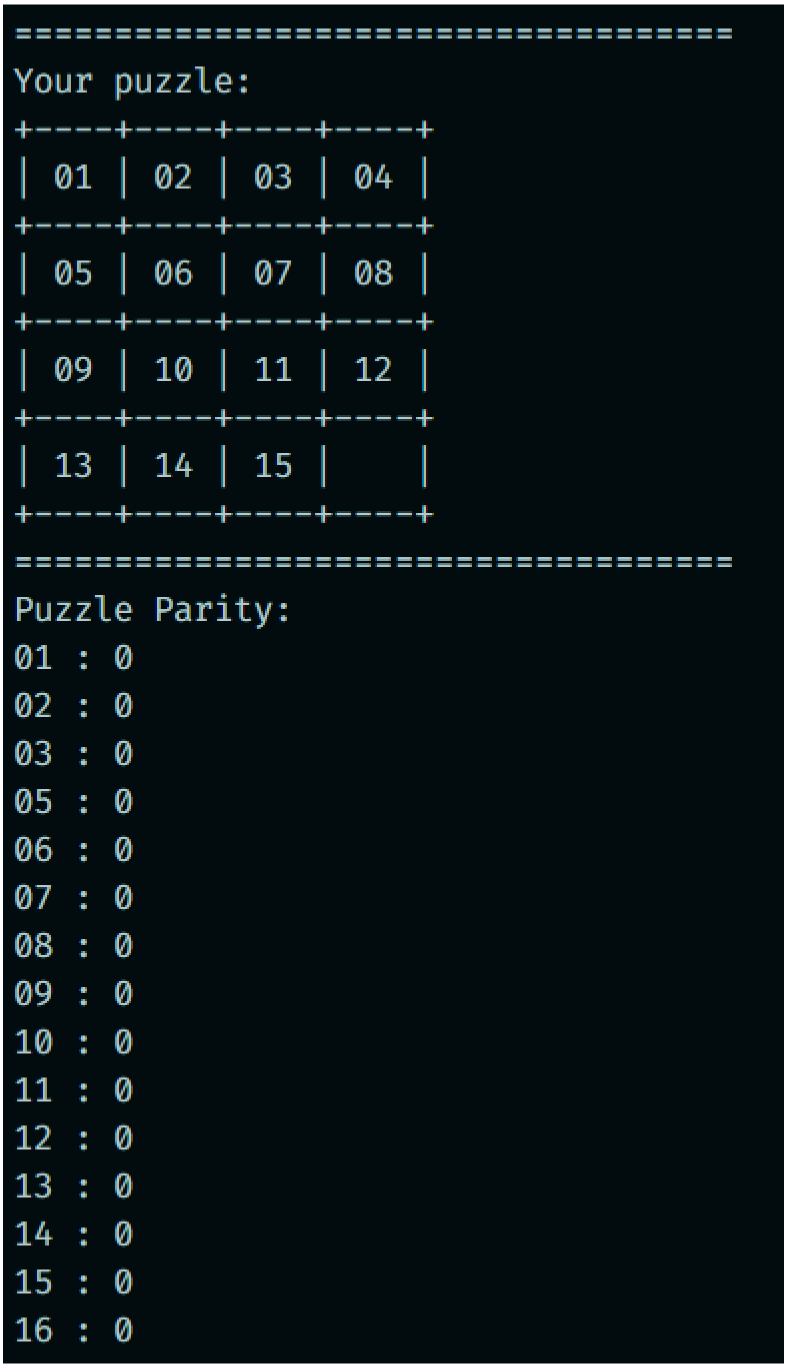
```
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 x
```

masukan

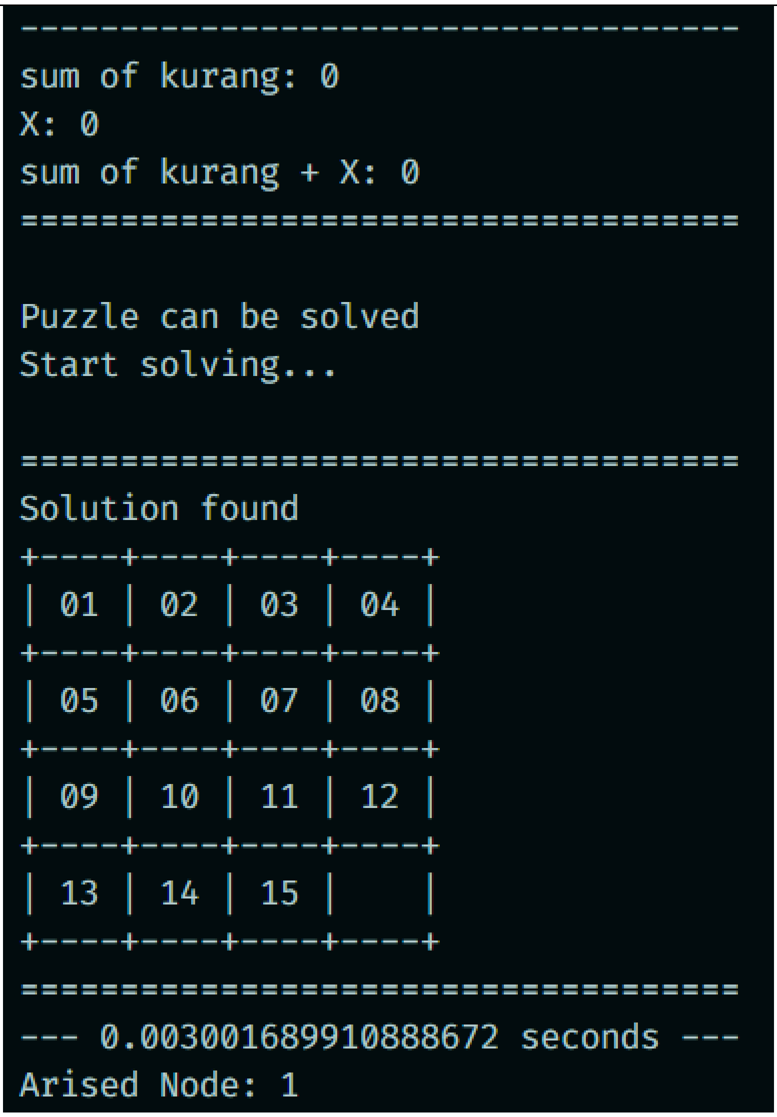


```
Welcome to the 15-Puzzle Solver Program by: Tri Sulton Adila (13520033)
Please enter the method to generate puzzle:
1. Random
2. Manual from file
Your choice: 2
1. Run all the file in the tests folder
2. Run a specific file
Your choice: 2
Please enter the file name: matriks3.txt
```

keluaran



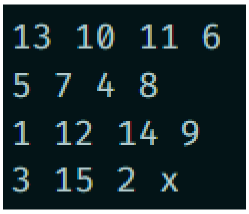
```
====================================
Your puzzle:
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 | 14 | 15 |    |
+----+----+----+----+
====================================
Puzzle Parity:
01 : 0
02 : 0
03 : 0
05 : 0
06 : 0
07 : 0
08 : 0
09 : 0
10 : 0
11 : 0
12 : 0
13 : 0
14 : 0
15 : 0
16 : 0
```



```
------------------------------------
sum of kurang: 0
X: 0
sum of kurang + X: 0
====================================

Puzzle can be solved
Start solving...

====================================
Solution found
+----+----+----+----+
| 01 | 02 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 | 14 | 15 |    |
+----+----+----+----+
====================================
--- 0.003001689910888672 seconds ---
Arised Node: 1
```

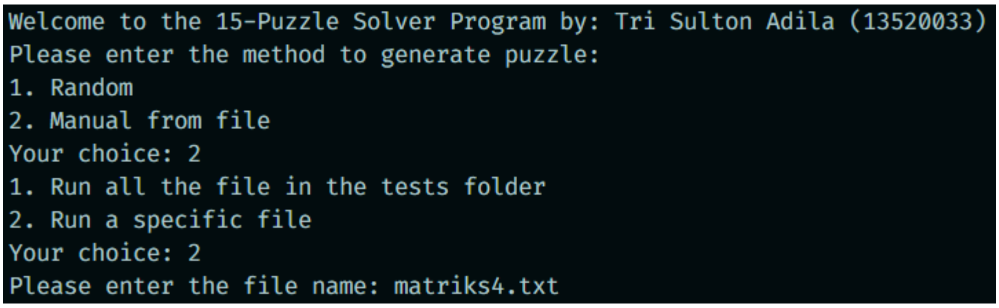
2. Not Reachable
   - matriks4.txt

matriks



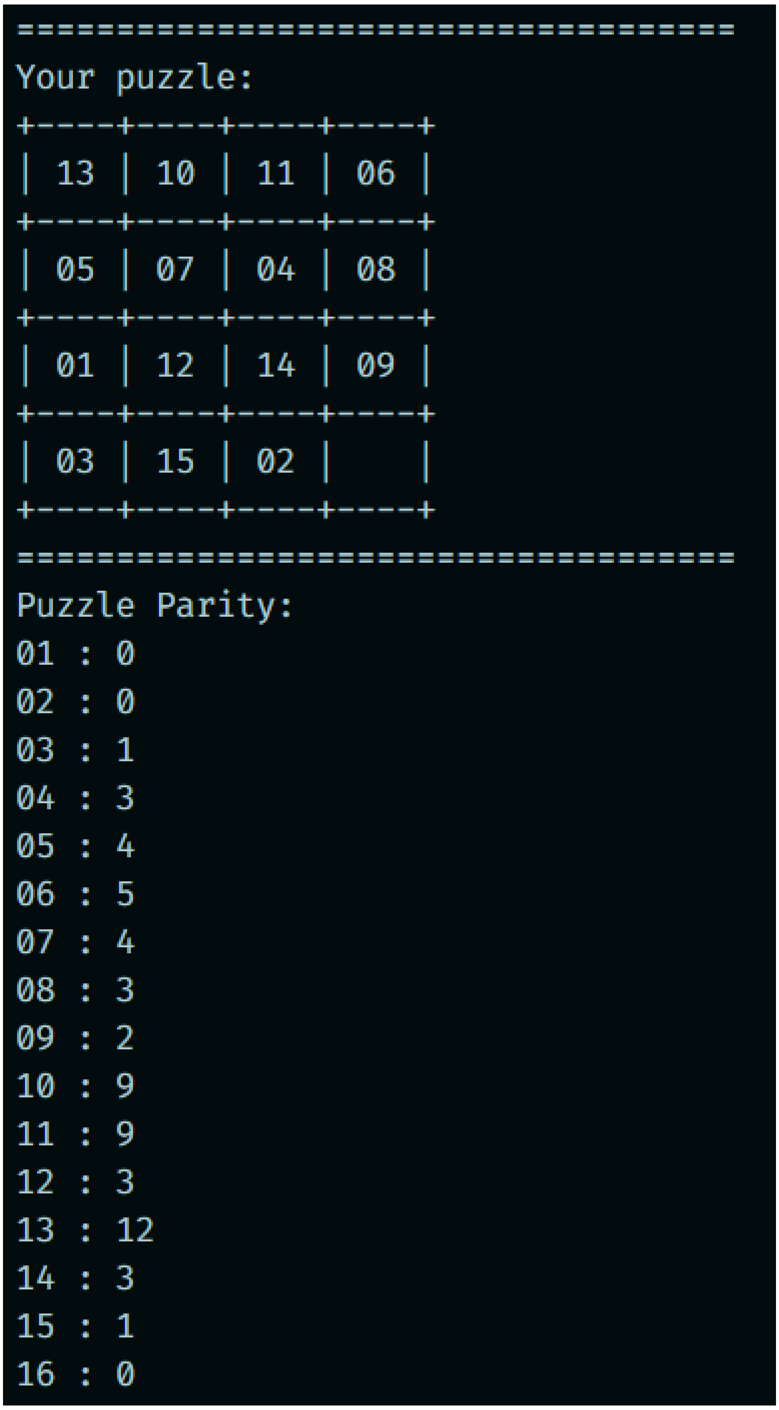
```
13 10 11 6
5 7 4 8
1 12 14 9
3 15 2 x
```

masukan

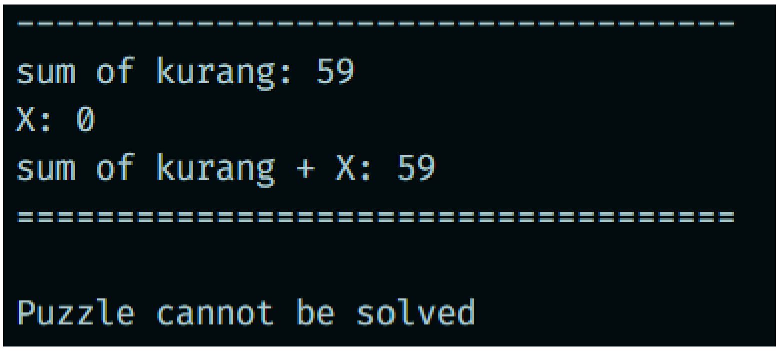


```
Welcome to the 15-Puzzle Solver Program by: Tri Sulton Adila (13520033)
Please enter the method to generate puzzle:
1. Random
2. Manual from file
Your choice: 2
1. Run all the file in the tests folder
2. Run a specific file
Your choice: 2
Please enter the file name: matriks4.txt
```

keluaran



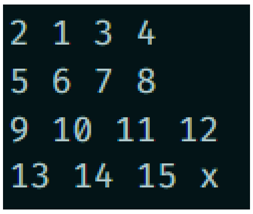
```
====================================
Your puzzle:
+----+----+----+----+
| 13 | 10 | 11 | 06 |
+----+----+----+----+
| 05 | 07 | 04 | 08 |
+----+----+----+----+
| 01 | 12 | 14 | 09 |
+----+----+----+----+
| 03 | 15 | 02 |    |
+----+----+----+----+
====================================
Puzzle Parity:
01 : 0
02 : 0
03 : 1
04 : 3
05 : 4
06 : 5
07 : 4
08 : 3
09 : 2
10 : 9
11 : 9
12 : 3
13 : 12
14 : 3
15 : 1
16 : 0
```



```
------------------------------------
sum of kurang: 59
X: 0
sum of kurang + X: 59
====================================

Puzzle cannot be solved
```

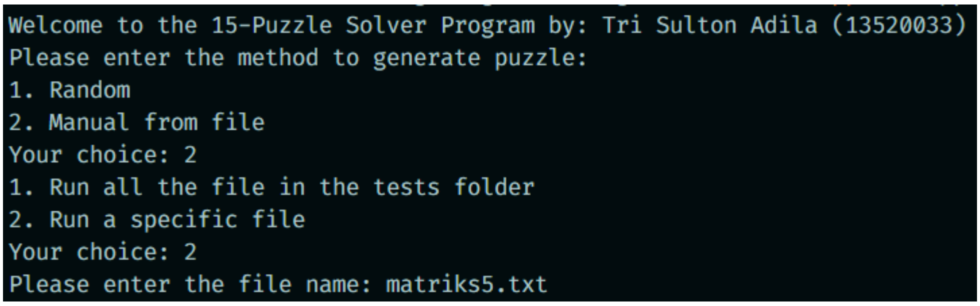
- matriks5.txt

matriks



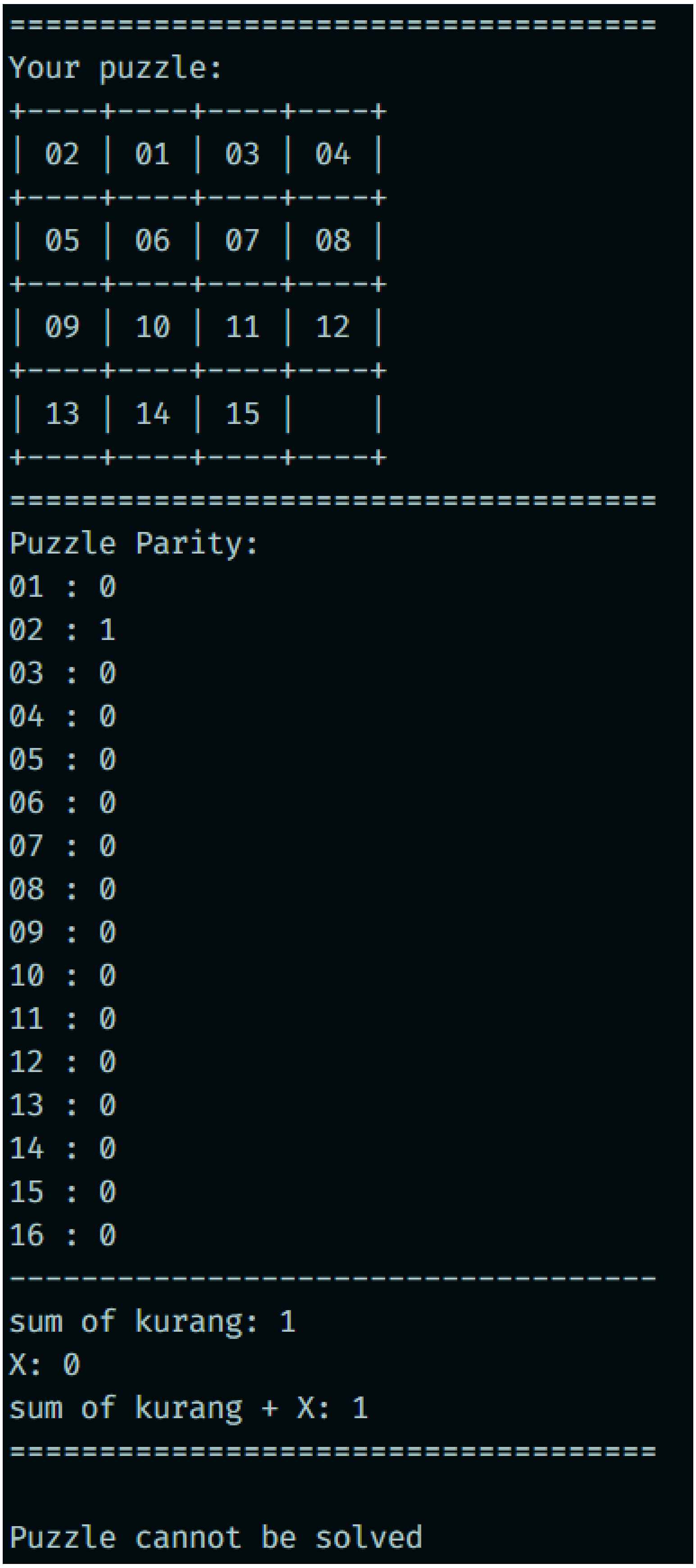
```
2 1 3 4
5 6 7 8
9 10 11 12
13 14 15 x
```

masukan



```
Welcome to the 15-Puzzle Solver Program by: Tri Sulton Adila (13520033)
Please enter the method to generate puzzle:
1. Random
2. Manual from file
Your choice: 2
1. Run all the file in the tests folder
2. Run a specific file
Your choice: 2
Please enter the file name: matriks5.txt
```

keluaran



```
====================================
Your puzzle:
+----+----+----+----+
| 02 | 01 | 03 | 04 |
+----+----+----+----+
| 05 | 06 | 07 | 08 |
+----+----+----+----+
| 09 | 10 | 11 | 12 |
+----+----+----+----+
| 13 | 14 | 15 |    |
+----+----+----+----+
====================================
Puzzle Parity:
01 : 0
02 : 1
03 : 0
04 : 0
05 : 0
06 : 0
07 : 0
08 : 0
09 : 0
10 : 0
11 : 0
12 : 0
13 : 0
14 : 0
15 : 0
16 : 0
------------------------------------
sum of kurang: 1
X: 0
sum of kurang + X: 1
====================================

Puzzle cannot be solved
```

## D. Alamat Github

https://github.com/3sulton/Tucil3_13520033

## E. Tabel

| Poin | Ya | Tidak |
|---|---|---|
| 1. Program berhasil dikompilasi | √ | |
| 2. Program berhasil running | √ | |
| 3. Program dapat menerima input dan menuliskan output | √ | |
| 4. Luaran sudah benar untuk semua data uji | √ | |
| 5. Bonus dibuat | | √ |