FAKULTETI I INXHINIERISË ELEKTRIKE DHE KOMPJUTERIKE

DEPARTAMENTI I INXHINIERISË KOMPJUTERIKE



✺ **Projekti**

➢ Sliding Puzzle Algorithm

**Data:** 22. 05. 2021

**Punoi:** Fitore Morina, Lirim Islami, Uran Lajçi

**Veglat e përdorura:** Microsoft Word, Google Docs, Vs code

**Sistemi Operativ:** Windows 10

**ID:** ,180714100087, 18071410065

**Gjuha Programuese:** Java

**Email:** fitore.morina11@student.uni-pr.edu, lirim.islami@student.uni-pr.edu, uran.lajci@student.uni-pr.edu

**Lënda:** Dizajni dhe Analiza e Algoritmeve

**Mentorët:** Avni Rexhepi, Dardan Shabani

Permbajtja

[**1.** **Hyrje** 3](#_Toc72121988)

[**2.** **Përshkrimi dhe spjegimi i Sliding Puzzle Algoritmit** 4](#_Toc72121989)

[**3.** **Perdorimet e Sliding Puzzle Algoritmit** 5](#_Toc72121990)

[**4.** **Implementimi dhe shpjegimi i kodit bazë** 6](#_Toc72121991)

[**4.1. Kodi Burimor** **Error! Bookmark not defined.**](#_Toc72121992)

[**4.2. Testimet** 7](#_Toc72121993)

[**4.2.1. Shpjegimi se si punon aplikacioni** 7](#_Toc72121994)

[**4.2.2. Krahasimi i algoritmeve** 9](#_Toc72121995)

[**4.2.3. Levizja e pozitave** 12](#_Toc72121996)

[**4.2.4. Validimet** 14](#_Toc72121997)

[**5.** **Referencat** 16](#_Toc72121998)

# **Hyrje**

Enigma e lëvizjes filloi në vitin 1880. Në pak muaj pas prezantimit të tij, njerëzit në të gjithë botën ishin të zhytur në përpjekje për të zgjidhur atë që u bë e njohur si Puzzle 15. Ai përbëhej nga një rrjet 4 me 4, me 15 katrorë të numëruar dhe një hapësirë të lënë bosh. Ideja ishte që të përziheshin numrat, dhe pastaj t'i rirregulloni ato në renditje numerike duke i rrëshqitur në mënyrë të njëpasnjëshme në hapësirën boshe. Ky enigmë e parë lëvizëse u shpik nga Noyes Chapman nga New York, i cili doli me këtë ide gjatë viteve 1870. Fabrikat nuk mund të vazhdonin me kërkesën për 15 Puzzle. Aq shumë njerëz u kapën nga këto enigma, saqë punëdhënësit ishin të shqetësuar për punëtorët duke injoruar detyrat e tyre për të punuar në enigmë. Ligjvënësit në Gjermani u panë duke u përpjekur t'i zgjidhnin ato. Mania lëvizëse e enigmave u përmend në këngët dhe shfaqjet popullore. Një dentist nga Worcester, Massachusetts i ofroi një sërë dhëmbësh dhe 100 dollarë çdo personi që mund të zgjidhte enigmën nga një rregullim i caktuar fillestar. Ai më vonë e rriti çmimin në 1.000 dollarë. Njerëzit në të gjithë vendin u përpoqën ta fitonin këtë shumë të madhe, por askush nuk ia doli. Zhvillimi tjetër në puzzles rrëshqitëse erdhi në 1909 kur Lewis W. Hardy shpiku enigmën e parë rrëshqitëse duke përdorur pjesë që ishin drejtkëndëshe. Ai e quajti atë Puzzle Pennant, duke i dhënë asaj një temë bejsbolli. Ai ishte i njohur gjithashtu si Puzzle i Babait .Ma's Puzzle, e cila pasoi në 1927, prezantoi dy pjesë në formë L, duke e bërë edhe më të vështirë zgjidhjen. Që kur kompjuteri filloi të luante një rol në hartimin dhe zgjidhjen e enigmave, shumëllojshmëria e enigmave është rritur. Qindra fotografi janë shndërruar në lojë lëvizëse dhe rregullimet gjithnjë e më të vështira të formave gjeometrike kanë sfiduar zgjidhësit.

# **Përshkrimi dhe spjegimi i Sliding Puzzle Algoritmit**

# **Perdorimet e Sliding Puzzle Algoritmit**

# **Implementimi dhe shpjegimi i kodit bazë**

Për të zgjedhur Sliding Puzzle me anë të një algoritmi duhet që të prezantohen gjendjet e enigmës si një graf, dhe pastaj të përdoren algoritmet për gjetje të shtegut si DFS (Depth-first search), BFS (Breadth-first search), Dijkstra, A\* etj. Algoritmet dallojnë nga mënyra se si arrijnë deri te zgjidhja e enigmës, disa algoritme janë më të shpejta se të tjerat.

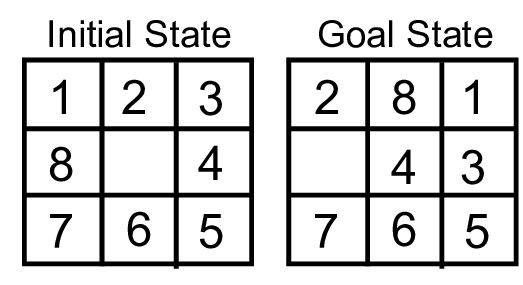


Figura 1. Gjendjet e enigmes

Enigma i ka dy gjendje kryesore, ato janë : Gjendja fillestare (Initial State) dhe Gjendja të cilën dëshirojmë ta arrijmë (Goal State). Mes këtyre dy gjendjeve mund të kemi zero (**kur enigma është e gjetur**) ose shume gjendje tjera.

Gjuha të cilën e kemi përdorur për të implementuar zgjidhjen e enigmës me anë të algoritmeve është Java. Ne programin që e kemi krijuar gjenden lëvizjet për të zgjedhur qdo enigmë lëvizëse (slidig puzzle) që mund të zgjidhet të rendit 3 ose 4 te matrices. Algoritmet me të cilat mundësohet zgjidhja e enigmës lëvizëse jane **A\* (Astar), BFS, DFS, dhe ID.**

Në IDE (Ne e kemi përdorur Eclipse, VSCode) duhet të ekzekutohet klasa PuzzleSolver.

## **4.2. Testimet**

### **4.2.1. Shpjegimi se si punon aplikacioni**



Figura 2. Pamja e pare qe shfaqet pasi te ekzekutohet programi.

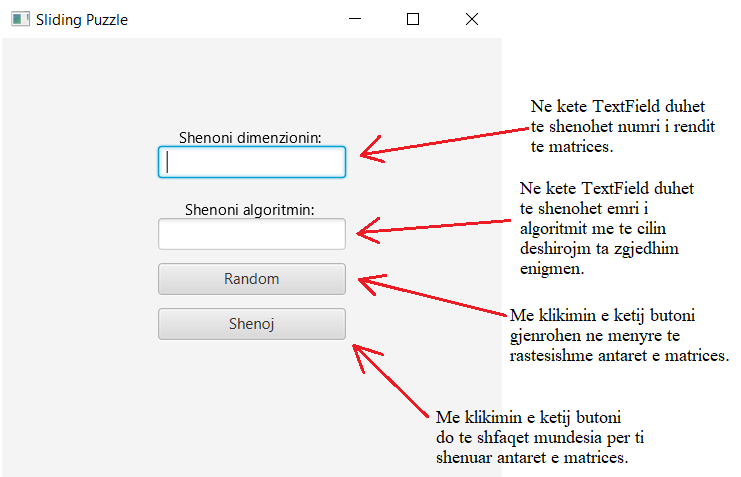


Figura 3. Shpjegimi se cka sherbejne fushat e paraqitura ne figure.

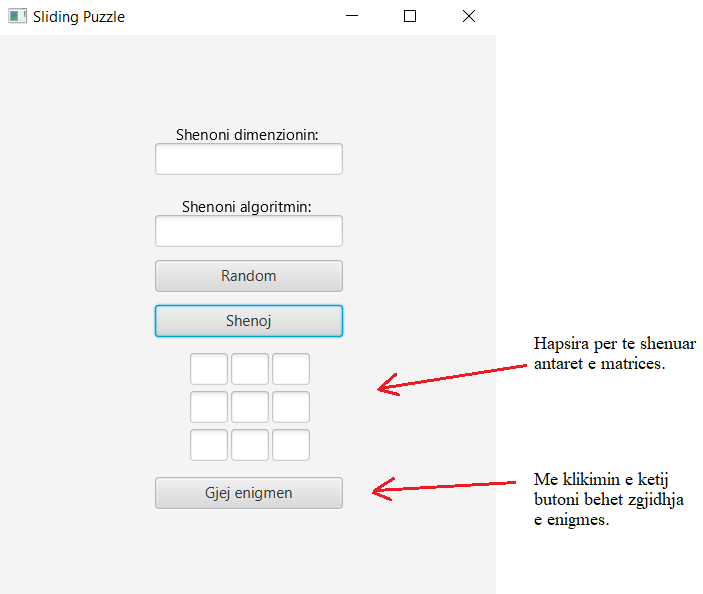


Figura 4. Pamja qe shfaqet pasi te klikohet butoni Shenoj.

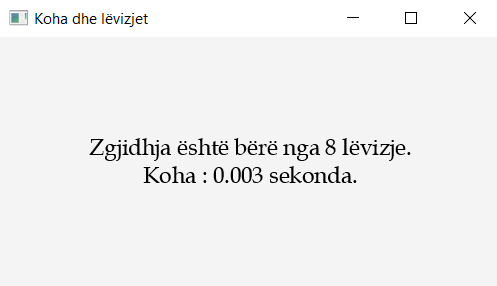


Figura 5. Pamja e pare qe shfaqet pasi te zgjidhet enigma.

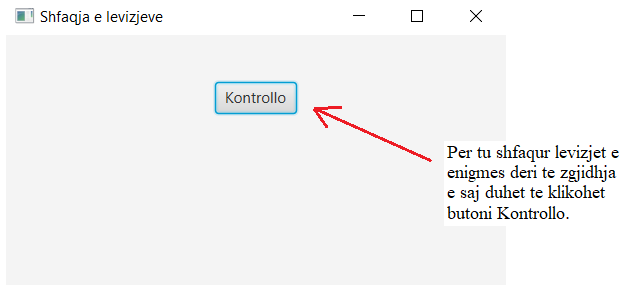


Figura 6. Pamja e dyte qe na shfaqet pasi te zgjedhet enigma.

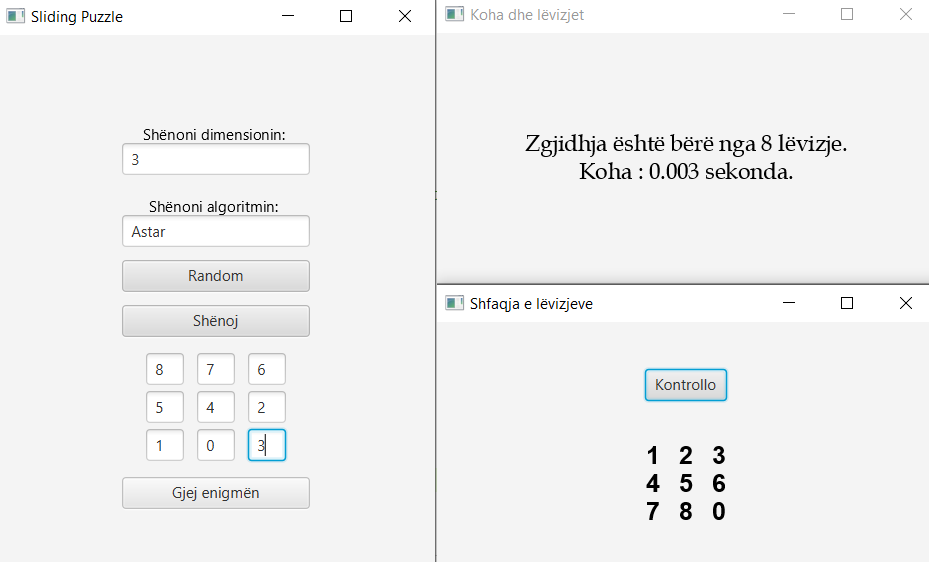


Figura 7. Pamja pas nje ekzekutimi te plote te programit.

### **4.2.2. Krahasimi i algoritmeve**

Figura 8. Koha ne sekoda qe ju duhen algoritmeve te ndryshme per te gjetur zgjidhjen e enigmes

Ne figuren 8 ne anen e majt te figures numrat paraqesin kohen e zgjidhjes se enigmes. Per tu fituar ky grafik jane bere 4 testime me 4 enigma te ndryshme, me te cilat eshte testuar koha e zgjidhjes se enigmes prej secilit algoritem. Nga kjo shohim se algoritmi A\*(Astar) e ka bere zgjidhjen me te shpejt te enigmave. Algoritmet BFS dhe DFS kane performuar perafersisht njejte, kurse Algoritmi ID ne 2 raste koha me te cilen e ka bere zgjidhjen e enigmes ka qene me e gjate.

Figura 9. Levizjet qe ju duhen algoritmeve per te gjetur zgjidhjen e enigmes

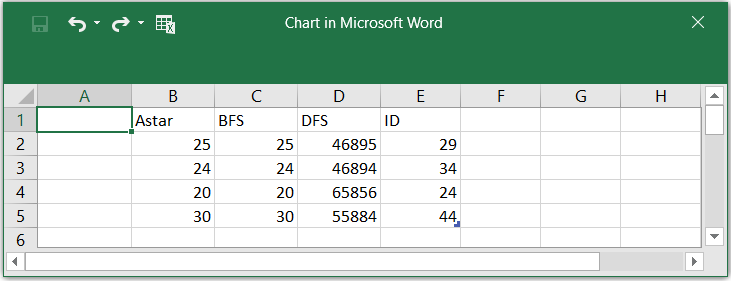


Figura 10. Numri i levizjeve qe eshte dhene si input per te fituar grafin ne figuren 9.

Nga figura 9 dhe figura 10 e shohim qe algoritmet Astar dhe BFS ne 4 raste kane zgjidhur enigmen me numer te njejte te levizjeve, kurse algoritmi ID ne te 4 rastet ka pasur numer me te madh te levizjeve per te zgjidhur algoritmin. Ndersa ajo qka mund te konkludojm eshte se algoritmi DFS ka patur nevoje per shume me shume levizje per te bere zgjidhjen e enigmes.

Figura 11. Numri i levizjeve i dhe numri i testimeve kane rritje lineare.

Ne anen e majt te figures 11 jane te paraqitur numri i levizjeve kurse ne anen e poshtme jane te paraqitur numri i testimeve.

Figura 12. Grafi i ndryshimi i kohes.

Nga grafiku ne figuren 12 e shohim se pas 23 testimeve koha per zgjidhjen e enigmes se njejt nga algoritmi i njejte ka tendence qe te ulet. Karakteristika kryesore qe verehet eshte se nga testimi i pare dhe testimi i dyte kemi reninen me te madhe te kohes per zgjidhjen e enigmes.

### **4.2.3. Levizja e pozitave**

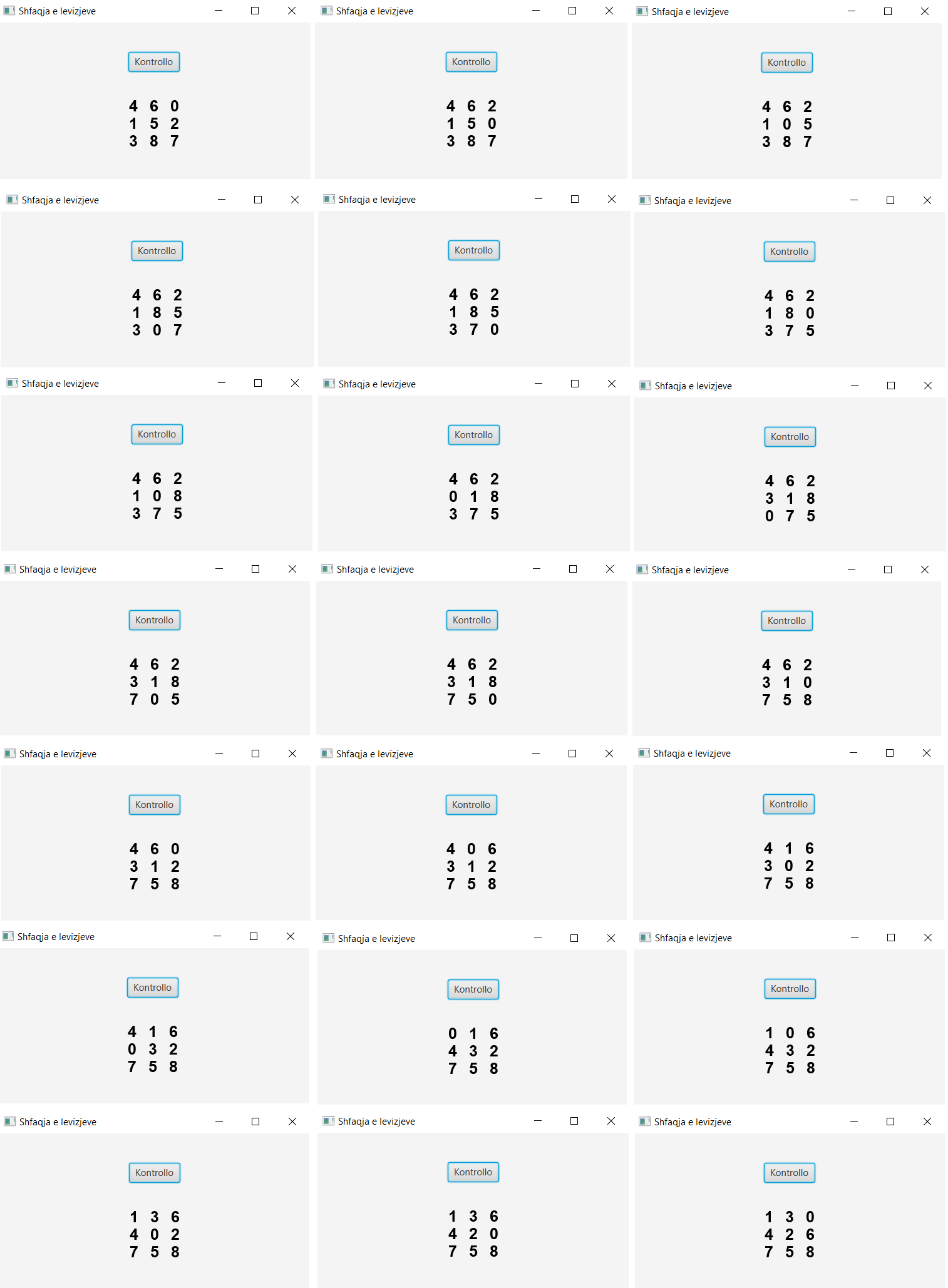


Figura .

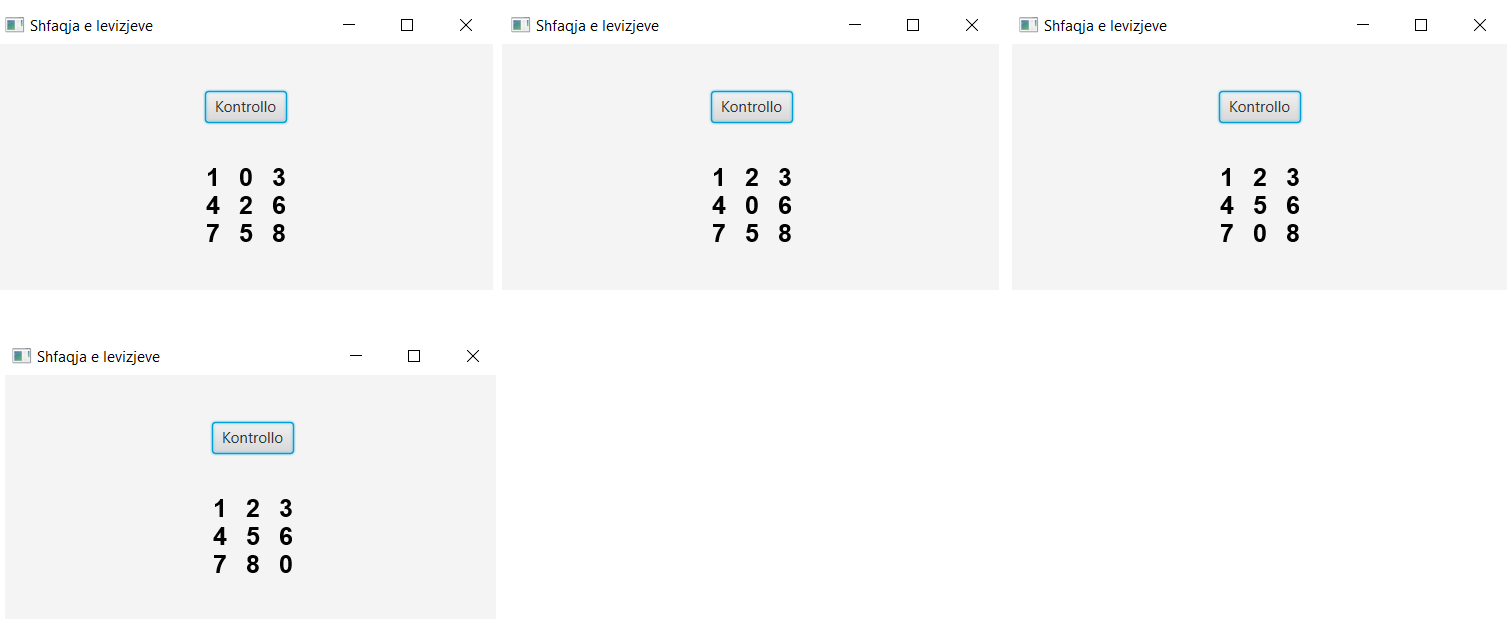


Figura 14. Duke klikuar ne butonin Kontrollo shohim hapat e zgjidhjes se enigmes.

### **4.2.4. Validimet**

Programi nuk mund te ekzekutohet si duhet nese:

1. nuk plotesohen te gjitha fushat ose ndonje prej tyre,
2. nese fushat plotesohen me te dhenat te gabuara.

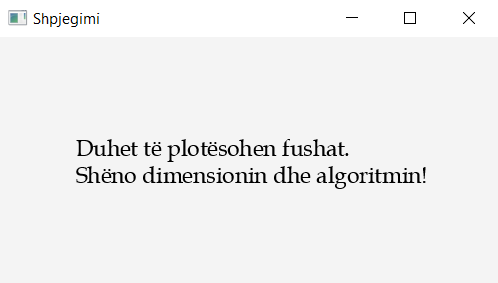


Figura 15. Pamja qe na shfaqet nese nuk plotesohen fusha e dimenzionit ose e algoritmit.

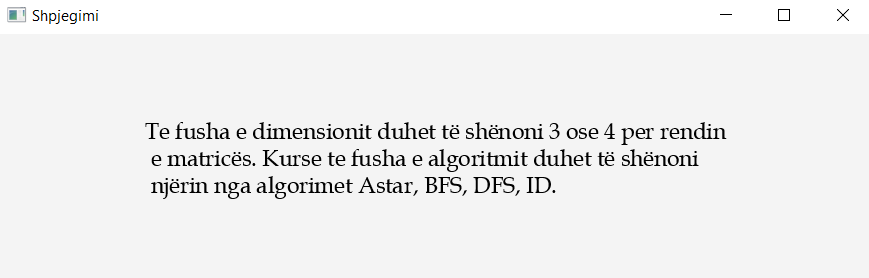


Figura 16. Pamja qe na shfaqet nese plotesohen fusha e dimenzionit dhe algoritmit me te dhena te gabuara.



Figura 17. Pamja qe na shfaqet nese nuk plotesohen te gjitha fushat e matrices.



Figura 18. Pamja qe na shfaqet nese fushat e matrices plotesohen me te dhena te gabuara.

# **Referencat**

1. Kodi burimor(pjesa e algoritmit) per zgjedhjen e enigmes eshte marre nga ky burim:

https://github.com/abalyani/Sliding-Puzzle-Solver