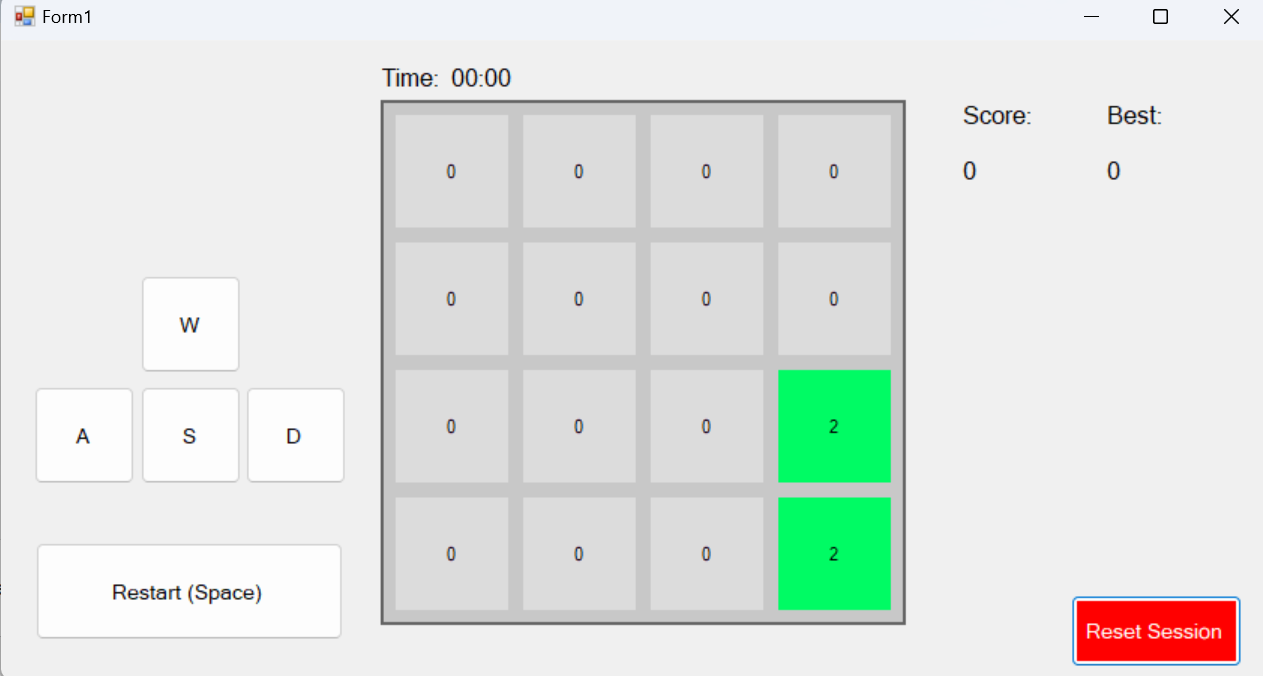
##1 Опис на апликацијата за визуелно програмирање 2048.

Играта 2048 е изработена во windows forms платформата. Таа е лесна да се започне да се игрa и се состои од создавање и изведување на алгоритми за поместување на квадрати од страна на корисникот.

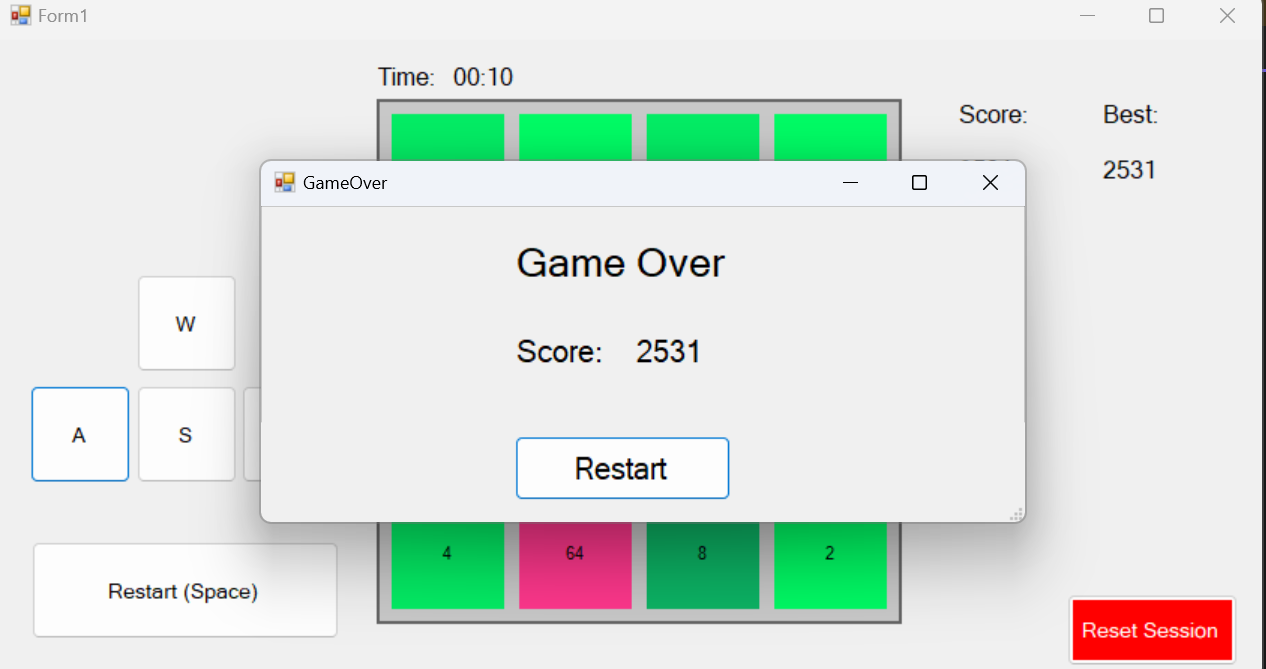
##2 Упатство за користање

Кога корисникот ја отвора програмата, тој ќе биде претставен со матрица од коцки во средината на екранот и копчиња за манипулација со матрицата. Копчињата може да се користат со кликање на глувчето или директно од тастатурата. 

Во матрицата случајно се доделани вредности на двa квадрати. Со кликање на едно од копчињата сите коцки се поместуваат на таа страна. Kога ќе се поместат ако се сретнат со друг квадрат која има иста вредност тие ќе се спојат, кога ќе се спојат тие си ја менуват бојата во однос на нивната вредност. Откако ќе се спојат на случајно место од матрициата се додава нов квадрат. Целта на играта е да се добие квадрат со што е можно поголема вредост.

Score е збирот на сите вредности во матрицата, но поголемите вредности носат повеќе поени. Best e најголемиот број на поени што корисникот ги освоил во ова сесија.

Time го брои времето откако корисникот го направил првото движење.

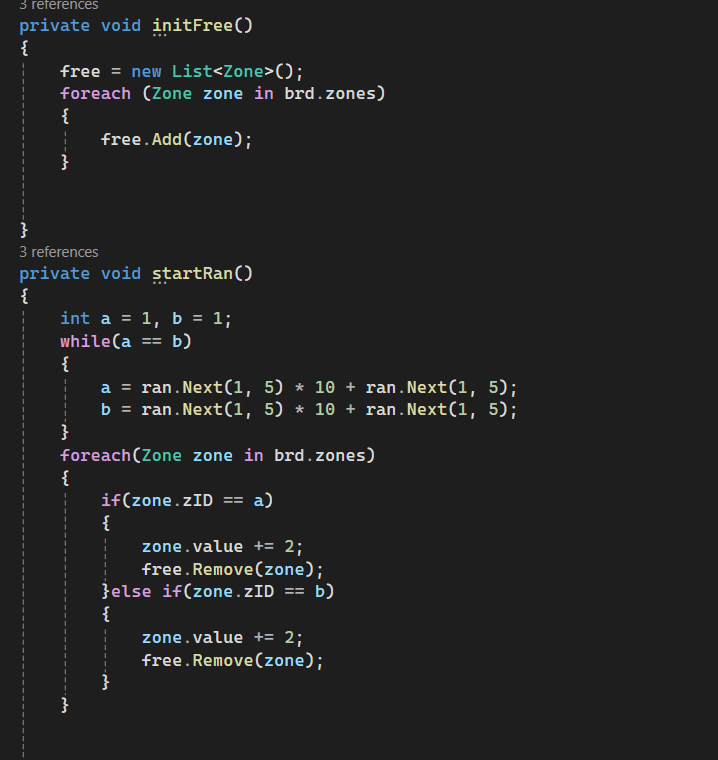


Кога корисникот ќе снема можни потези играта завшува. Се појавува прозорец кој ги прикажува неговите поени и дава опција да се започне одново.

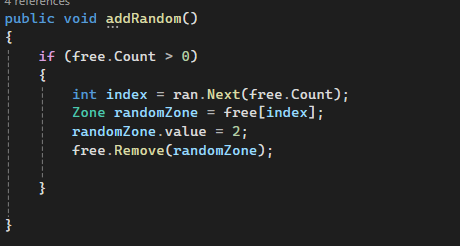
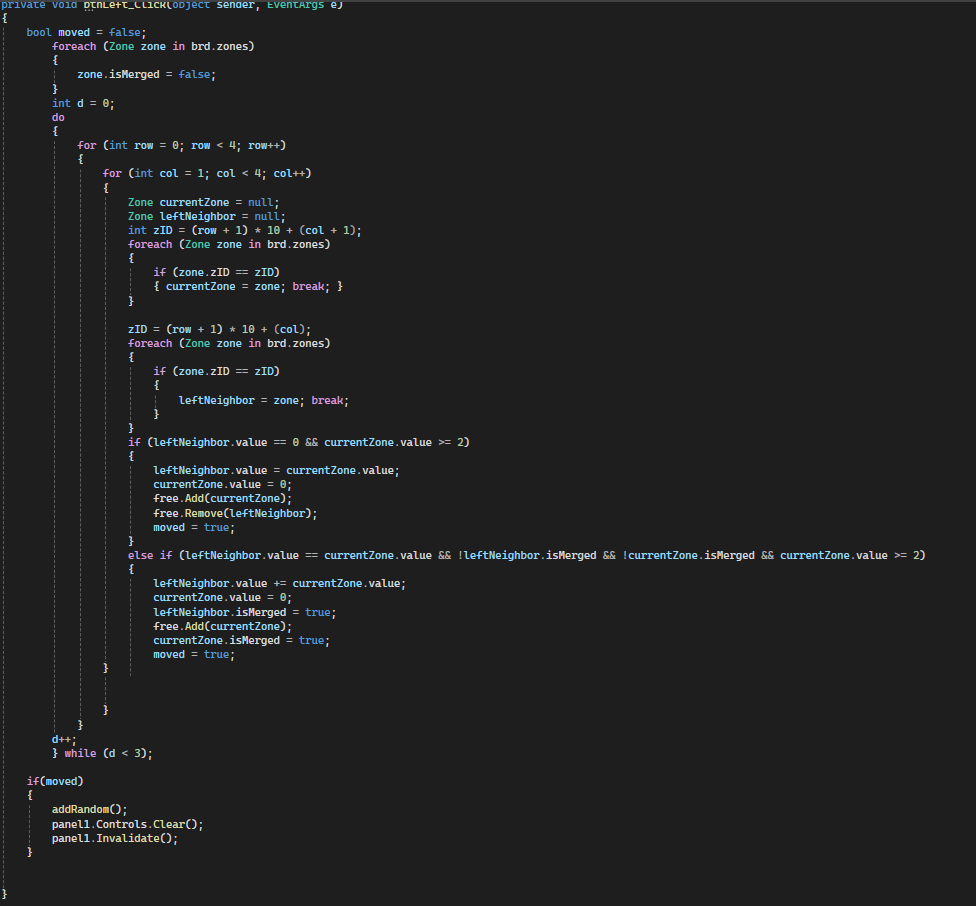
##3 Објаснување на кодот

###3.1 Главен фајл (Form1.cs)

Form1 е каде се наоѓа главната логика на апликацијата.



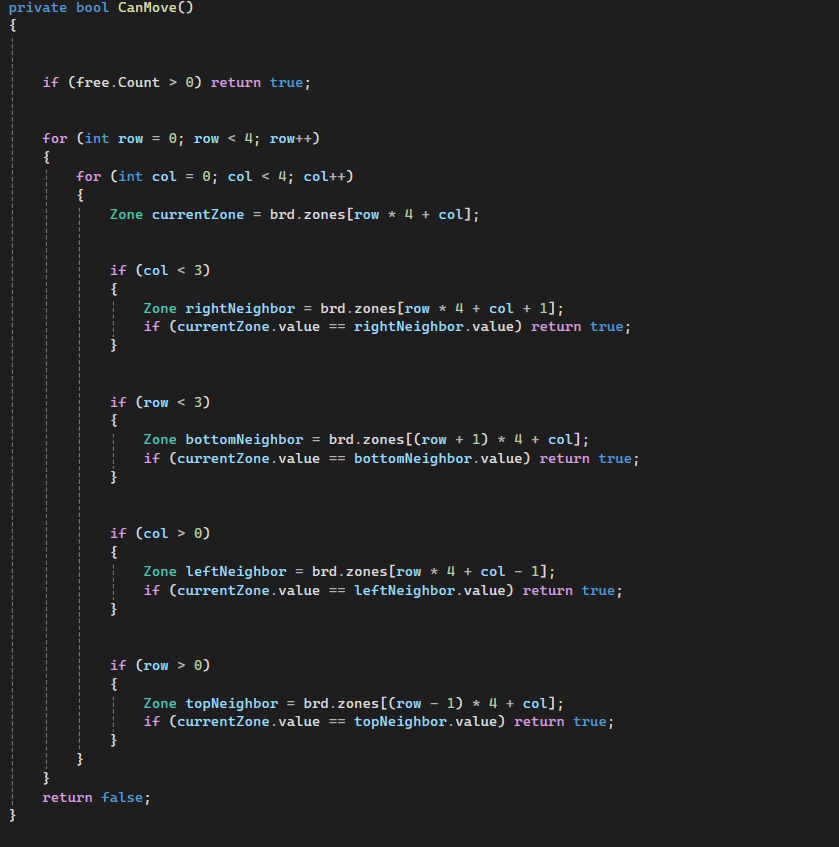
Со initFree() се додават сите квадрати во матрицата во листа на слободни квадрати(без вредност). Со startRan() се создаваат првите два квадрати при кое прво се утврдува дека тие не се исти.



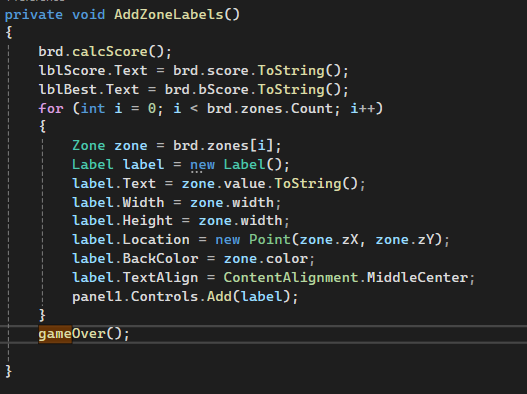
Овој дел е кодот за движење налево. Иницијално се создават двa објекта со вредност null кои ќе ги изминат сите редици и колони. Ако currentZone има вредност еднаква или поголема од нула проверувадали leftNeighbour лево од него нема вредност или има иста вредност како него. Ако се исполнети условите, тој ќе биде додаден на листата од слободни квадрати и неговата вредност ќе стане нула. Со moved се гледа дали настанало поместување, ако има некоја промена повторно ќе се исцрта панелот panel1 каде што се наоѓат квадратите.

Со addRandom() на случаен квадрат кој нема вредност му е зададена вредност 2.

Со слична логика се изведуват поместувањана во другите насоки и проверката дали е можно да се изведе поместување во иднина.

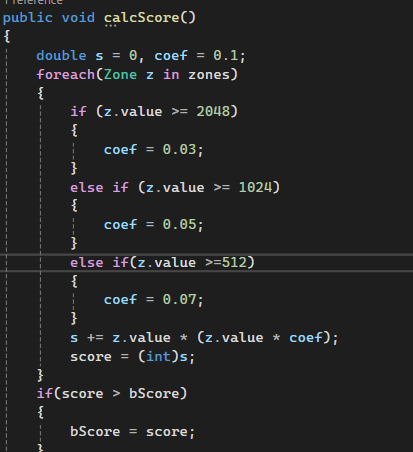


Овде се проверува дали тие имат сосед во кој што можат да се поместат. Ако не,се враќа вредност false.



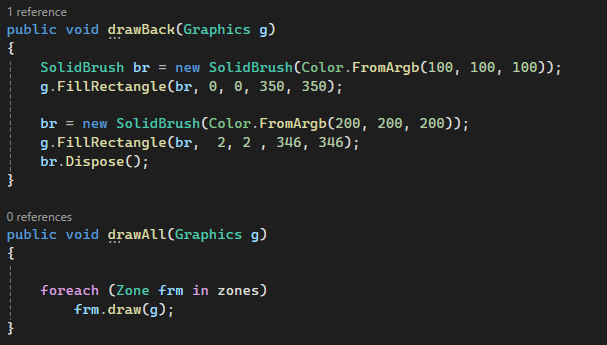
Со AddZoneLabels() се создават броевите во квадратите и се менува нивната боја, тие ги добиват своите особини како ширина и позиција од класата board.

###3.2 board



Во board пресметуват поените кои ги добил корисникот со calcScore(). Се користи коефициент кој што се намалува со зголемување на вредноста . Со calcScore исто така се проверува дали сегашните поени ја надминуват најголемата вредност од ова сесија.

Во board класата e зачувана листата не квадрати и е дефинирано исцртувањето на полето на матрицата. Со drawAll() се повикува исцртувањето на сите квадрати,



###3.3 Zone

Со calcColor() ја добиваме бојата на објектите. Бојата се менува со менувањето на вредноста на квадратот. Ако квадратот нема вредност тогаш тој е сив.

Со calcPosition() се добива позицијата на објектот во однос на panel1.

