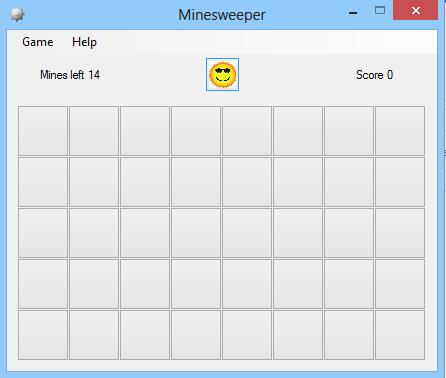
Универзитет “Св. Кирил и Методиј”

Факултет за информатички науки и компјутерско инженерство – Скопје

Проектна задача по предметот

Визуелно програмирање

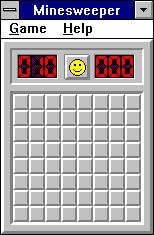
Изработиле:

Атанасовски Ѓорѓи 115007

Атанасовски Кристијан 115008

Скопје, 2013

Опис на проектот

Играта која ја имплементираме се вика Minesweeper и е една од најпопуларните игри од своето време. Прв пат се појавува заедно со оперативниот систем “Windows 3.1” во 1992. Иако интерфејсот на играта изгледа доста едноставен, оваа игра бара доста знаење и концентрација, но некогаш е во прашање и среќата. Во споредба со оригиналот нашиот проект нема претрпено многу промени. Правилата за игра се исти, треба да се отворат сите полиња на кои нема мини, а доколку кликнете на некое поле со мина сите полиња со мини се отвораат и играта е завршена. Промените кои ги направивме се тоа што дозволено е да се поставуваат неограничен број на знаменца и поените се калкулираат врз основа на тоа колку полиња сте отвориле бидејќи основните нивоа е потешки од оригиналната игра и потребно е премногу време за концентрација. Како ќе приметите дека сте креирале успешна стратегија за играње? За таа помош доколку сте погодиле со некое знаменце некоја мина доколку погодите друга мина на тоа место ќе се појави едно насмеано човече со шлем , а доколку сте промашиле а тоа било бројка на местото на знаменцето ќе се појави сликичка со прецртана бомба. Но доколку ви се смачи од основните нивоа тука е и опцијата сами на креирате игра, но овојпат доколку ставите да има само едно слободно поле ќе треба да го најдете. Другите алатки немаат некоја посебна функционалност освен копчето со сонце кое служи за креирање нова игра он истото ниво.

Решение на проблемот

За да го решиме проблемот прво креиравме класа “obj” која ни претставува едно поле/копче на екранот. Во неа се чуваат информации за тоа дали е отворено копчето, дали во моментот има знаме или прашалник на него , вредноста доколку е број, и променлива за копчето. Освен податоците кои ги чуваме во оваа класа се содржат уште неколку функции кои играат главна улога за да се избегне buffering при играње и да не се зафаќа меморија со постојано креирање и бришење на објекти. Основниот конструктор “public obj(int size, int y, int x, string n,int pozi,int pozj)“ кој ги прима повеќето аргументи како име на копчето , големина , локација на формата и локација во матрицата. Во него се додава и “MouseEventHandler” а исто така и оваа функција се содржи во објектот. Функцијата за клик на копчето нема некоја посебни пресметки туку користи повикување на други функции во зависност од потребата користи 4 променливи за да се заштити од поклопување на функции меѓу кои дали е отворено копчето и дали е овозможен лев/десен клик. Од останатите функции тука е и

“public void OpenAllMines() “ која има пристап до главната матрица во формата и ги отвора сите мини доколку погодиме една, а исто така ги менува објектите на кои имаме ставено знаменце , “public void setImage()“ повикана од левиот клик доколку копчето е број поставува слика на него , “public void objSet(string s) “ откако е креиран објектот прима само име и се сетира според името и “public void objReset()“ служи за да се постави објектот на почетна состојба.

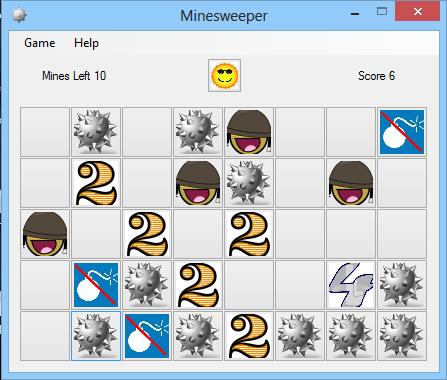
Главната форма содржи малку посложени функции и динамички додадени елементи. Креирана е променлива “public enum Level “ која ни помага за полесно креирање нова игра со само повикување на функцијата откако ќе го одбереме нивото. Главниот елемент во формата е матрица 20х20 од “obj” објекти која откако ќе се алоцира во неа се додаваат сите објекти кои ќе бидат достапни за играње. Покрај тоа се чуваат и моменталниот број на полиња по ширина и висина , моментален број на мини кои остануваат исти се додека не го промените нивото.

Функции за работа и иницијализација на “minefield”

Во главната форма има голем број помошни функции кои се повикуваат по потреба. Тука е функцијата “public int surround(int i, int j,int x,int y)“ која има улога да го врати бројот на мини кои се наоѓаат околу полето на кое сакаме да поставиме вредност. Функцијата “public void cleanfield() “ која ги изминува сите употребени објекти и ги враќа во почетната состојба. Со помош на оваа функција се избегнува чекање, кочење и ред други проблеми кои се појавуваат при иницијализирање на работната површина на формата. Функцијата “private List<E> ShuffleList<E>(List<E> inputList) “ која ни помага при поставување на мините, за да се избегне чекање при поставување на рандом место на бомбите на работната површина.

Функцијата “public static void openCleanArea(int i1, int j1 ) “ , оваа функција е повикана од страна на објект од класата “obj “ доколку кликнеме на празно поле. Тука рекурзивно се отвораат сите локални полиња кои се празни или имаат број во големина на полињата за игра и притоа сите полиња кои ќе ги отвориме им ставаме статус “opened“ за да не се повикува непотребно функцијата кај поминатите полиња. За полињата со бројка само се прикажува сликата, а празните полиња се менуваат во “disabled“. Рекурзијата се повикува на сите осум страни со тоа што се проверува дали е на некој од краевите за да не излезе од своите граници

Функцијата “public void newGame()“ која е најчесто повикувана, се повикува при секој повик за менување на ниво или за нова игра. Нема никакви калкулации во неа но служи за да повика функција со која се иницијализира работната површина во зависност од нивото кое е одбрано и му ги задава параметрите.

Најглавната функција за крај “public void canvas(Level lvl,int sizeX,int sizeY,int mines)“. Оваа функција повикана од “newGame()“ ја подесува формата како и локацијата на сите елементи кои се во неа зависно од зададените параметри што ни е од голема помош бидејќи не зависи толку од нивото колку од параметрите и со тоа можеме да креираме каква сакаме работна површина. Во неа прво се креираат две листи. Во првата се додаваат индексите на елементите кои ќе ги користиме и потоа со помошната функција ги мешаме . Од неа во втората листа ги додаваме првите неколку елементи колку што ќе има мини во површината која ја користиме на тој начин брзо и без многу пресметки и оптоварување на процесорот ги имаме позициите на мините . ги поставуваме копчето за брз пристап на игра на средина и двете лабели од страните. Ја изминуваме работната површина за да ги додадеме мините чии локации ги земаме од втората листа. Доколку преминуваме од едно во друго ниво се изминува уште еднаш целата матрица од крајот кон почетокот за да се тргнат вишокот копчиња на екранот и на крај ги поставуваме полињата кои треба да имаат вредност број со тоа што ја изминуваме целата матрица поле по поле и доколку е празно поле ја повикуваме функцијата “surround(i,j,sizeY,sizeX) “ и ја поставуваме вредноста на објектот со бројката на мини околу него бидејќи кога ќе биде кликнат тој сам си ја поставува сликата со помошна функција. И играта може да започне!!!