Vidzemes Augstskola

Inženierzinātņu fakultāte

2. kurss

Programmas izstrāde C++ vidē

Spēle “1024”

Darba autors:

Kristaps Naglis

Valmiera, 2019

Saturs

[1. Programmas apraksts 4](#_Toc8893424)

[1.1. Vispārīgs apraksts 4](#_Toc8893425)

[1.2. Funkciju bloku apraksts 4](#_Toc8893426)

[1.2.1. Main.cpp 4](#_Toc8893427)

[1.2.2. Game1024.cpp 4](#_Toc8893428)

[Funkcijas 5](#_Toc8893429)

[Public 5](#_Toc8893430)

[**void newGame();** 5](#_Toc8893431)

[**void printWelcomeScreenUI();** 5](#_Toc8893432)

[**void printUI();** 5](#_Toc8893433)

[**void applyMove(int);** 5](#_Toc8893434)

[Private 5](#_Toc8893435)

[**void setColor(int);** 5](#_Toc8893436)

[**pair<int, int> generateUnoccupiedPosition();** 5](#_Toc8893437)

[**void generateNewNumber();** 5](#_Toc8893438)

[**void countScore();** 6](#_Toc8893439)

[**bool isBoardFull();** 6](#_Toc8893440)

[**bool isGoalAchived();** 6](#_Toc8893441)

[**bool canMakeMove(int, int, int, int);** 6](#_Toc8893442)

[**void retryScreenUI();** 6](#_Toc8893443)

[**void congratulationsScreenUI();** 6](#_Toc8893444)

[Funkciju hierarhiskais pārskats 7](#_Toc8893445)

[2. Izmantotās bibliotēkas 8](#_Toc8893446)

[2.1. <iomanip> 8](#_Toc8893447)

[2.2. <conio.h> 8](#_Toc8893448)

[3. Programmas darbības demonstrācija 10](#_Toc8893449)

[4. Bibliogrāfiskais saraksts 13](#_Toc8893450)

# Programmas apraksts

## Vispārīgs apraksts

Spēle “1024” ir matemātiska skaitļu spēle ar 4x4 liels laukumu, kurā atrodas pāra skaitļi. Spēles mērķis ir pārvietot šos skaitļus laukuma robežās, apvienojot vienādos skaitļus. Sākumā ir 2 skaitļi, kas jāapvieno. Tos apvienot var uzbīdot vienu virs otra. Pārvietot skaitļus iespējams 4 virzienos – uz augšu, pa kreisi, uz leju, pa labi. Kad tiek apvienoti divi skaitļi, rodas šo abu skaitļu summa. Pēc katra gājiena parādās jauns skaitlis, kas var būt vai nu 2, vai 4. Spēle tiek pārtraukta divos gadījumos:

* spēlētājs spēli ir uzvarējis, kad tiek izveidots skaitlis 1024;
* spēlētājs ir piepildījis visu laukumu ar skaitļiem un vairs nevar pārbīdīt skaitļus;

Paralēli tiek skaitīti punkti, kas spēlētājam tiek attēloti spēles beigās.

## Funkciju bloku apraksts

### Main.cpp

Šī funkcija ir programmas ieejas punkts un lielākoties atbild par to, lai lietotājs varētu ievadīt komandas spēles kontrolēšanai, kā arī tā atbild par spēles darbību - loga atsvaidzināšanu pēc katras lietotāja ievadītās komandas.

Main funkcijas sākumā tiek izveidots klases Game1024 objekts, ar kuru zemāk tiek realizēta šīs klases funkciju izpilde. Tiek arī inicializēts nejaušu skaitļu ģenerators, kas izmanto datora pulksteni, kā *seed*, lai nejaušo skaitļu ģenerēšanā nerastos paredzami rezultāti.

### Game1024.cpp

Klase Game1024 atbild par spēles loģiku un interfeisa nodrošināšanu. Sākot ar spēles laukuma izdruku, beidzot ar punktu skaitīšanu un skaitļu pārbīdīšanu.

#### Funkcijas

##### Public

###### **void newGame();**

Sagatavo laukumu un rezultātus jaunai spēlei.

###### **void printWelcomeScreenUI();**

Izdrukā spēles sākuma ekrāna informāciju.

###### **void printUI();**

Izdrukā galveno spēles informāciju – spēles laukumu, iegūtos punktus, spēles noteikumus.

###### **void applyMove(int);**

Spēles visietilpīgākā funkcija. Atbild par sekmīgu skaitļu summēšanu, pārvietošanu pa laukumu, notikumu pārbaudi.

Argumenti:

* *direction – INTEGER tipa mainīgais, kas nosaka lietotāja ievadīto virzienu, uz kuru pārbīdīt skaitļus laukumā.*

##### Private

###### **void setColor(int);**

Maina konsoles izvada burtraksta krāsu un burtraksta fona krāsu.

Argumenti:

* *colorChoice – INTEGER tipa mainīgais, kas nosaka burtraksta un fona krāsu.*

###### **pair<int, int> generateUnoccupiedPosition();**

Funkcija atrod un atgriež brīvu pozīciju spēles laukumā.

**Atgrieztā vērtība**: pair INTEGER tipa skaitļi – *row* un *column.*

###### **void generateNewNumber();**

Funkcija ģenerē nejaušu skaitli, no dotā masīva, kurā ir divas vērtības – 2 un 4 – un šo ģenerēto vērtību ievieto spēles laukumā.

###### **void countScore();**

Funkcija aprēķina lietotāja spēles rezultātu.

###### **bool isBoardFull();**

Funkcija pārbauda, vai spēles laukums ir pilns ar skaitļiem.

Atgrieztā vērtība: true vai false

###### **bool isGoalAchived();**

Funkcija pārbauda, vai spēlētājs ir sasniedzis mērķa rezultātu – 1024.

Atgrieztā vērtība: true vai false

###### **bool canMakeMove(int, int, int, int);**

Funkcija pārbauda, vai ir iespējams pārvietot skaitļus kādā noteiktā virzienā. Pārbauda, vai skaitļi ir pie spēles laukuma malas un vai pārbīdot tos, tie *neizkāptu* ārpus robežām.

Argumenti:

* row – INTEGER tipa mainīgais, kas norāda pašreizējo rindas pozīciju laukumā;
* column – INTEGER tipa mainīgais, kas norāda pašreizējo kolonnas pozīciju laukumā;
* nextRow - INTEGER tipa mainīgais, kas norāda nākamo rindas pozīciju laukumā;
* nextColumn - INTEGER tipa mainīgais, kas norāda nākamo kolonnas pozīciju laukumā;

Atgrieztā vērtība: true vai false

###### **void retryScreenUI();**

Izdrukā spēles atkārtošanas ekrānu, kad spēlētājs zaudē.

###### **void congratulationsScreenUI();**

Izdrukā spēles uzvaras ekrānu, kad spēlētājs sekmīgi pievar spēli.

##### Funkciju hierarhiskais pārskats

class Game1024 {

public:

void newGame();

void printWelcomeScreenUI();

void printUI();

void applyMove(int);

private:

void setColor(int);

std::pair<int, int> generateUnoccupiedPosition();

void generateNewNumber();

void countScore();

bool isBoardFull();

bool isGoalAchived();

bool canMakeMove(int, int, int, int);

void retryScreenUI();

void congratulationsScreenUI();

};

# Izmantotās bibliotēkas

Šajā kursa darbā tika izmantotas sekojošas, kursā neapskatītas bibliotēkas:

#include <iomanip>

#include <conio.h>

#include <windows.h>

## <iomanip>

Šī bibliotēka tiek izmantota, lai varētu vizuāli pievilcīgi noformatēt spēles laukumu. No šīs bibliotēkas tiek izmantota funkcija setw(4). Šī funkcija nodrošina atstarpi starp punktiem, kas nemainās, kad tiek ievadīts skaitlis. Funkcijas arguments “4” norāda uz atstarpes platumu, kas ir 4 *whitespace* vienības liels.



## <conio.h>

Šī bibliotēka tiek izmantota, lai lietotājam nospiežot jebkuru pogu uz klaviatūras, nebūtu nepieciešams apstiprināt ievadu ar ENTER taustiņu, bet gan ievads tiktu apstiprināts automātiski, uzreiz pēc ievada veikšanas. Šādu funkcionalitāti nodrošina \_getch() funkcija, kas ievadā saņemot jebkuru char, automātiski apstiprina to.

* 1. <windows.h>

Windows.h bibliotēka nodrošina consoles izvada noformēšanas iespēju, izmantojot dažādas fontu un to fonu krāsas. Par šo funkcionalitāti rūpējas vairākas koda līnijas, kas iekļautas kopīgā funkcijā setColor(int colorChoice).

void Game1024::setColor(int colorChoice) {

HANDLE hConsole;

int k = colorChoice;

hConsole = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(hConsole, k);

}

Funkcija kā argumenta vērtību saņem INTEGER tipa mainīgo, kas nosaka, kāda krāsa būs fontam un kāda būs fona krāsa, ja vispār. Krāsas tiek noteiktas pēc sekojoša principa:

* Skaitlis no 0 līdz 15 ir krāsas. Izvēloties tikai vienu skaitli nomainīsies fonta krāsa

|  |  |  |
| --- | --- | --- |
| **ID** | **Krāsa** | **Gaišums** |
| 1 | Zila | 50% |
| 2 | Zaļa | 50% |
| 3 | Tirkīza | 50% |
| 4 | Sarkana | 50% |
| 5 | Rozā | 50% |
| 6 | Dzeltena | 50% |
| 7 | Balta | 50% |
| 8 | Pelēka | 50% |
| 9 | Zila | 100% |
| 10 | Zaļa | 100% |
| 11 | Tirkīza | 100% |
| 12 | Sarkana | 100% |
| 13 | Rozā | 100% |
| 14 | Dzeltena | 100% |
| 15 | Balta | 100% |

* Lai izvelētos gan fonta, gan fona krāsas, jāveic sekojoša formula:
  + Gala kombinācija = fonta krāsa + fona krāsa \* 16;

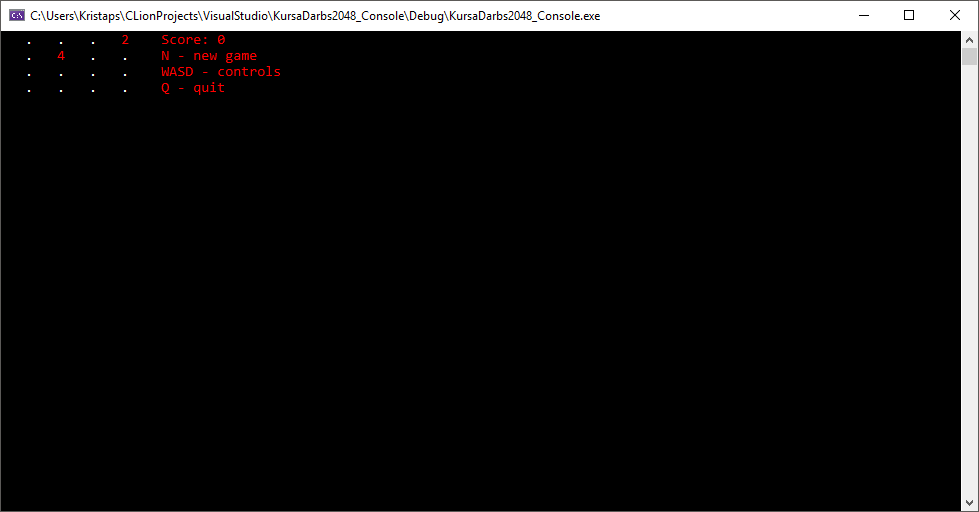
# Programmas darbības demonstrācija

Startējot programmu, lietotājs tiek novirzīts uz spēles sākuma ekrānu.



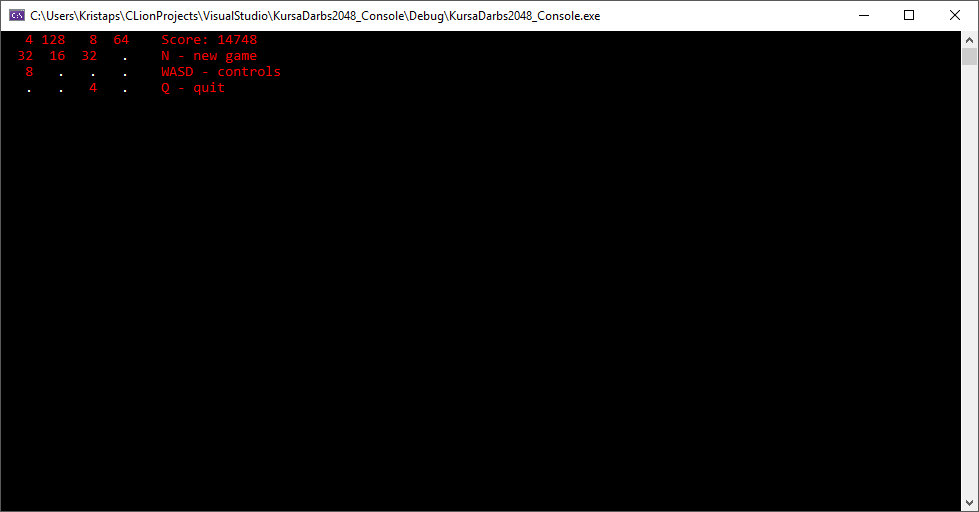
1. *attēls – spēles starta ekrāns*

Starta ekrānā nospiežot ENTER taustiņu, lietotājs tiks novirzīts uz spēles sākumu.



1. *attēls – spēles sākums*

Lietotājs spēli var kontrolēt ar WASD taustiņiem, tādā veidā bīdot skaitļus pa spēles laukumu un vienādos skaitļus kombinējot.



1. *attēls – iesākta spēle*

Kad viss laukums tiek piepildīts ar skaitļiem un lietotājs mēģina pārbīdīt skaitļus, kur tas nav iespējams, spēle ir zaudēta un lietotājs tiek novirzīts uz zaudējuma ekrānu, kur tiek norādīts iegūto punktu skaits un iespēja sākt spēli pa jaunam. 

1. *attēls – zaudētas spēles ekrāns*

Taču kad tiek sasniegts skaitlis 1024 uz spēles laukuma, spēlētājs tiek novirzīts uz spēles pieveikšanas ekrāna, kur tiek parādīts spēlētāja iegūtais punktu skaits.



1. *attēls – uzvarētas spēles ekrāns*

# Bibliogrāfiskais saraksts

<http://www.fredosaurus.com/notes-cpp/misc/random.html>

[https://www.daniweb.com/programming/software-development/code/216345/add-a-little-color-to-your-console-text#](https://www.daniweb.com/programming/software-development/code/216345/add-a-little-color-to-your-console-text)

<http://www.asciitable.com/>

<https://stackoverflow.com/questions/13553872/header-file-for-pair-stl>

<https://www.youtube.com/watch?v=tGWhk47ZIm4>

<https://stackoverflow.com/questions/4053837/colorizing-text-in-the-console-with-c>

<http://www.edparrish.net/common/cppdoc.html>

<https://www.geeksforgeeks.org/pair-in-cpp-stl/>

<http://www.cplusplus.com/reference/iomanip/>

<https://en.wikipedia.org/wiki/Conio.h>