



**AKADEMIA GÓRNICZO-HUTNICZA**

Dokumentacja do projektu

## **Gra Sudoku**

z przedmiotu

**Języki programowania obiektowego**

tu wpisać kierunek i rok studiów

*Jan Gądek*

wtorek 13:00

prowadzący: Rafał Frączek

22.01.23

# 1. Opis projektu

Projekt gry Sudoku. Użytkownik poprzez panel menu wybiera opcję rozgrywki(rodzaj poziomu trudności) i rozpoczyna rozgrywkę gry Sudoku. Gracz wpisuje liczby poprzez wpisanie numeru wiersza i kolumny a następnie wartości. Wybory gracza są bieżąco sprawdzane przez algorytm czy dana liczba już znajduje się w wierszu, kolumnie lub małym kwadracie 3x3.

## 2. Project description

Sudoku game project. Player through menu panel sets difficulty level and starts Sudoku. User sets values by typing row and column number and then algorithm tells user if this value already is in a row or column or small cue 3x3;

## 3. Instrukcja użytkownika

Rozgrzywka sudoku polega na wypełnieniu planszy o wymiarach 9x9, poprawnymi liczbami z zakresu 1-9. Plansza dodatkowo podzielona jest na 9 mniejszych kwadratów 3x3. Liczby nie mogą się powtarzać w wierszach, kolumnach oraz mniejszych kwadratach. Użytkownik na wstępie dostaje możliwość rozpoczęcia poprzez wpisanie „start” lub zakończenia rozgrywki „exit”. Następnie wybiera poziom trudności (1-9) i gra zostaje rozpoczęta. Zostaje wyświetlona tablica sudoku z 0 w pustych miejscach. Użytkownik wprowadza numer wiersza i numer kolumny a następnie zostaje poproszony o wprowadzenie wartości. Po każdej próbie wstawienia liczby w nieodpowiednim miejscu zostaje poinformowany. Po poprawnym uzupełnieniu sudoku gracz dostaje informacje zwrotną o sukcesie i ponowną możliwość gry.

## 4. Kompilacja

Dla tego projektu wystarczy standardowa kompilacja.

## 5. Pliki źródłowe

Projekt składa się z następujących plików źródłowych:

- *Sudoku\_1.cpp* – deklaracja oraz implementacja gry Sudoku

## 6. Zależności

W projekcie wykorzystano następujące dodatkowe biblioteki:

- Brak

## 7. Opis klas

W projekcie utworzono następujące klasy:

- Sudoku – reprezentuje konto w systemie bankowym.
  - void copy(int sudokuboard[N][N]) - metoda pobierająca tablice bool() od „pustego” sudoku,
  - void print(int sudokuboard[N][N]) - metoda wyświetlająca uzupełnianą tablice[9][9] sudoku ,
  - bool checkwin(int sudokuboard[N][N]) - metoda sprawdzająca czy dana tablica została poprawnie wypełniona,
  - bool checkColumn(int sudokuboard[N][N], int k, int value) - metoda sprawdzająca czy wartość „value” pojawiła się w kolumnie „k” tablicy,
  - bool checkRow(int sudokuboard[N][N], int w, int value) - metoda sprawdzająca czy wartość „value” pojawiła się w wierszu „w” tablicy,

- `bool checkcue(int sudokuboard[N][N], int initial_row, int initial_column, int value)` - metoda sprawdzająca czy wartość "value" pojawiła się w małym kwadracie,
- `void set(int sudokuboard[N][N], int w, int k, int value)` – metoda wpisująca liczby do tablicy sudoku po spełnieniu warunków, oraz wyświetlaniu informacji o statusie wpisania liczby .
- Zestaw – klasa zawierająca wygenerowane plansze sudoku do uzupełnienia

## 8. Zasoby

W projekcie wykorzystywane są następujące pliki zasobów:

- `brak`

## 9. Dalszy rozwój i ulepszenia

W przyszłości można zaimplementować automatyczne generowanie planszy sudoku N rozmiaru oraz wprowadzić ustawianie liczb sudoku poprzez „strzałki”.

## 10. Inne

`brak`