



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

---

Dokumentacja do projektu

# Connect Four

z przedmiotu

## Języki programowania

Kierunek: Elektronika I

*Kacper Filipek*

*czwartek 14:40*

Prowadzący: Rafał Frączek

Kraków, 24 maja 2022

# 1. Opis projektu

---

Projekt jest realizacją gry w "Czwórki" (ang. "Connect Four") w C++, z interfejsem TUI (t.j. interfejs używający znaków specjalnych i kolorów do rysowania interfejsu użytkownika w konsoli).

## 2. Project description

---

The project is a realisation of the game "Connect Four" made in C++ with TUI interface (an interface using special characters for drawing UI in the terminal)

## 3. Instrukcja użytkownika

---

Po włączeniu programu wyświetla się menu główne z następującymi opcjami:

- Play - rozpoczyna grę w bieżącymi ustawieniami
- Load game - umożliwia wczytanie pliku z zapisem gry
- Options - otwiera menu umożliwiające zmianę ustawień gry
- Quit game - wychodzi z programu

Wciśnięcie klawisza Escape spowoduje pauzę w grze i wyświetlenie się menu z następującymi opcjami:

- Resume - wznowia przebieg gry (alternatywnie można ponownie wcisnąć Escape)
- Save game - zapisuje stan gry do pliku
- Main menu - kończy grę i przechodzi do głównego menu bez zapisu

Wybranie "Load game" wyświetla użytkownikowi pole tekstowe, do którego należy wpisać nazwę pliku z zapisem gry. W przypadku błędu w odczycie pliku, wyświetli się komunikat powiadamiający o wystąpieniu błędu.

W menu "Options" można zmienić następujące parametry:

- Board width - szerokość planszy do gry (domyślnie 7)
- Board height - wysokość planszy do gry (domyślnie 5)
- Victory condition - liczba żetonów w sekwencji potrzebna do wygrania gry (domyślnie 4)

## 4. Kompilacja

---

Program został napisany na systemy operacyjne z rodziny Linux, chociaż powinien on działać na Windowsie, ponieważ kod nie używa żadnych zależnych od platformy plików nagłówkowych. Program używa systemu CMake do budowania projektu. Można skompilować go na dwa sposoby:

- Sposób 1:
  1. Wejść do folderu build/
  2. Wykonać polecenie `cmake ..`
- Sposób 2:
  1. Z katalogu głównego wywołać skrypt `./bld.sh`. Ze względu na fakt, że skrypt wywołuje program `make`, może on nie działać na Windowsie.

Po zbudowaniu powinien się plik wykonywalny o `./build/connect-four`. Z uwagi na fakt, że ścieżki do zasobów są wpisane w programie relatywnie do głównego katalogu, to program wykonywalny powinien z niego wywoływany.

## 5. Pliki źródłowe

---

W tym punkcie należy opisać wszystkie pliki źródłowe (.cpp, .h) w projekcie. Należy podać nazwę każdego pliku oraz informację o tym co się w nim znajduje. Na przykład: Projekt składa się z następujących plików źródłowych:

- board.h, board.cpp – deklaracja oraz implementacja klasy Board,
- game.h, game.cpp – deklaracja oraz implementacja klasy Game,
- menu.h, menu.cpp – deklaracja oraz implementacja klasy Menu.
- extras.h, extras.cpp – deklaracja oraz implementacja funkcji pomocniczych.
- color.h – definicje procesora nazw kolorów do użycia w funkcjach biblioteki ncurses.
- main.cpp – główny plik z implementacją funkcji main.

## 6. Zależności

---

W projekcie wykorzystano następujące dodatkowe biblioteki:

- ncurses – biblioteka do interakcji z emulatorem terminala, pozwala tworzyć zaawansowane interfejsy konsolowe:  
<https://invisible-island.net/ncurses/>

## 7. Opis klas

---

W projekcie utworzono następujące klasy:

- Menu - klasa reprezentująca menu główne programu
  - Menu()
  - void Draw()
  - void Update()
  - void Start() - rozpoczyna pętlę klasy Menu
  - void key\_handler() - funkcja obsługująca inputy od użytkownika w menu głównym
  - void next\_option() - ustawia następną opcję w menu
  - void prev\_option() - ustawia poprzednią opcję w menu
- Game - klasa reprezentująca stan i właściwy przebieg rozgrywki
  - Game() - domyślny konstruktor klasy Game
  - void Start() - ustawia parametry i rozpoczyna pętlę gry
  - void set\_parameters(GameParameters parameters) - ustawia parametry gry na podstawie podanego argumentu
- Board - klasa reprezentująca planszę do gry
  - Board() - domyślny konstruktor, ustawiający domyślne parametry
  - Board(uint8\_t board\_rows, uint8\_t board\_columns) - konstruktor parametryczny ustawia rozmiar planszy na board\_rows × board\_columns
  - uint16\_t get\_columns() - zwraca liczbę kolumn planszy
  - uint16\_t get\_rows() - zwraca liczbę wierszy planszy
  - void set\_dimensions(uint16\_t rows, uint16\_t columns) - ustawia wymiary planszy

## 8. Zasoby

---

W projekcie wykorzystywane są następujące pliki zasobów:

- assets/ – katalog zawierający dodatkowe zasoby do gry.  
Struktura katalogu:
  - logo1.txt, logo2.txt – pliki zawierające tekstowe logo pojawiające się na ekranie startowym.

## **9. Dalszy rozwój i ulepszenia**

---

## **10. Inne**

---