

Politechnika Śląska w Gliwicach

Wydział Automatyki, Elektroniki i Informatyki



# **Podstawy Programowania Komputerów**

**Mastermind**

---

Autor: Dominik Dziembała

Prowadzacy: dr inż. Piotr Pecka

Rok akademicki: 2015/2016

Kierunek: informatyka

Rodzaj studiów: SSI

Semestr: 2

Termin laboratorium / ćwiczeń: czwartek 8:30-9:00

Grupa: 7

Sekcja: 2

Data sporządzenia sprawozdania: 29.08.2016

---

## 1. Temat

Gra mastermind (wymagane GUI).

## 2. Analiza projektowanie

### 2.1. Algorytmy, struktury danych

- **Struktury danych:** Wykorzystam tablice statyczną, której będę przechowywał informacje o wybranym kolorze kuli i o jej pozycji podczas ustalania kodu, jak również informację o odgadywanej kuli.
- **Algorytmy:** Wykorzystam zmodyfikowany algorytm sortowania bąbelkowego do sprawdzenia, czy odgadywana kombinacja kul zgadza się z kombinacją ustawioną jako kod.

### 2.2. Analiza obiektowa.

Aby stworzyć graficzny interfejs użytkownika posłużę się biblioteką GTK+ za pomocą której utworzę okno, kontenery oraz przyciski.

## 3. Specyfikacja zewnętrzna

### 3.1. Obsługa programu

W menu głównym gry mamy do wyboru 3 przyciski: Nowa gra, Instrukcja oraz Wyjście. Przycisk Wyjście powoduje zamknięcie programu. Przycisk Instrukcja wyświetla okno z zasadami gry. Przycisk Nowa gra wywołuje kolejne menu, w którym możemy wybrać jeden z dwóch trybu rozgrywki: 1vs1 oraz Odgadnij kod. Tryb 1vs1 to gra dla dwóch graczy, gdzie pierwszy gracz ustala kod a drugi zgaduje, a następnie zamieniają się rolami. Tryb Odgadnij kod to tryb w którym komputer ustala kod z kul, a gracz musi odgadnąć tą kombinację.

### 3.2. Komunikaty

Po kliknięciu w przycisk Instrukcja wyświetla się nam komunikat, w którym są zawarte wszystkie reguły jak należy grać w Mastermind. Po ukończeniu gry w trybie Odgadnij kod mogą wyświetlić się dwa różne komunikaty, jeden o treści, że udało nam się odgadnąć kombinację, drugi, że nie daliśmy rady. Z kolei w trybie 1vs1 po ukończeniu odgadywania kombinacji wyświetla się komunikat informujący gracza który odgadywał kod czy mu się udało czy nie, a po zamianie ról wyświetla wynik rozgrywki, czyli który gracz wygrał lub czy jest remis.

## 4. Specyfikacja wewnętrzna

## 4.1. Zmienne

Zmienna	Objaśnienie
GtkWidget *okno;	Tworzy okno programu.
GtkWidget *obraz;	Zawiera obrazek znajdujący się nad głównym menu programu.
GtkWidget *info[4];	Zawiera komuniaty z informacjami dla graczy podczas trybu 1vs1.
GtkWidget *konte;	Kontener główny do którego jest wrzucana reszta elementów.
GtkWidget *konte2;	Kontener który znajduje się w oknie do przewijania.
GtkWidget *scrolled;	Okno do przewijania.
GtkWidget *game;	Przycisk Nowa gra.
GtkWidget *instr;	Przycisk Instrukcja.
GtkWidget *exitt;	Przycisk Wyjście.
GtkWidget *kolor[6];	Przyciski z kolorami.
GtkWidget *kod[4];	Przyciski, które symbolizują jaki kolor znajduje się na której pozycji w sekwencji kodu.
GtkWidget *wybor[4];	Przyciski, który po naciśnięciu aktywują przyciski "kolor" i przypisują kolor przyciskowi kod.
GtkWidget *proba[9][4];	Przyciski, które symbolizują jaki kolor znajduje się na której pozycji w sekwencji zgadywania kodu.
GtkWidget *wproba[9][4];	Przyciski, który po naciśnięciu aktywują przyciski "kolor" i przypisują kolor przyciskowi proba.
GtkWidget *sprawdze[9][4];	Przyciski informujące za pomocą czarnej i białej kuli o poprawności sekwencji.
int *tproba[4];	Liczbowa interpretacja wybranego koloru w przycisku proba.
GtkWidget *startk;	Przycisk kończący ustalanie kodu.
GtkWidget *sprawdzz;	Przycisk sprawdzający zgodność kolorów z przycisków "proba" i "kod".
int *tkod[4];	Liczbowa interpretacja wybranego koloru w przycisku kod.
int y, j, i, a, b, c, d, f, aa, bb, cc, dd, ff, e;	Określają pozycję przycisków w "konte2".
int test;	Służy jako indeks.
int runda[3];	Informuje o rundzie.
int kontrola;	Informuje, czy wszystkie przyciski "kod" mają już wartość.
int test2;	Służy jako indeks.
int pomoc;	Przywraca na stałe wartość ostatniemu wybranemu elementowi z "tkod" na stałe.
int omin[5];	Zapisuje w tej tablicy informacje które kolory znajdują się na swoim miejscu.
int podo[4][2];	Zapisuje w tej tablicy informacje które kolory znajdują się na w kodzie, ale nie są na swoim miejscu.

## 4.2. Funkcje

Funkcja	Objaśnienie
void gtk1(int argc, char *argv[])	Funkcja tworzy okno programu, menu główne gry i wstawia obrazek.
void grasz(GtkWidget *widget, tmp *data)	Funkcja tworzy menu wyboru trybu gry.
void graszz(GtkWidget *widget, tmp *data)	Funkcja rozpoczyna rozgrywkę Odgadnij kod, losuje liczby od 1 do 6 i ustawia je jako kod oraz tworzy przyciski z kolorami.
void instruk(GtkWidget *widget, tmp *data)	Funkcja tworzy okno dialogowe z instrukcją.
void gras(GtkWidget *widget, tmp *data)	Funkcja rozpoczyna rozgrywkę 1vs1, tworzy przyciski z kolorami, jeden z graczy wybiera kod, a następnie po jego ustawieniu może przejść do dalszego etapu przyciskiem Start.
void wybierz_kod(GtkWidget *kolorek, tmp *data)	Funkcja dezaktywuje pozostałe 3 przyciski tabeli wybor i aktywuje przyciski z kolorami do wyboru.
void wybierz1(GtkWidget *widget, tmp *data)	Funkcja dla wybranego elementu tabeli kod ustawia kolor czerwony.
void start(tmp *data)	Funkcja sprawdza czy wszystkie elementy z tabeli kod posiadają już kolor.
void rozgrywka (GtkWidget *widget, tmp* data)	Funkcja rozpoczyna etap odgadywania kodu przez drugiego gracza.
void rozgrywka2 (GtkWidget *widget, tmp* data)	Funkcja rozpoczyna etap odgadywania kodu stworzonego przez program przez gracza.
void polaczeniaa(tmp *data)	Funkcja dla trybu Odgadnij kod tworzy połączenie dla przycisków z kolorami i przycisków z pojedynczego wiersza z tabeli wproba.
void polaczenia(tmp *data)	Funkcja dla trybu 1vs1 tworzy połączenie dla przycisków z kolorami i przycisków z pojedynczego wiersza z tabeli wproba.
void wybierz_proba(GtkWidget *kolorek, tmp *data)	Funkcja dezaktywuje pozostałe 3 przyciski tabeli wproba i aktywuje przyciski z kolorami do wyboru.
void wybierz11(GtkWidget *widget, tmp *data)	Funkcja dla wybranego elementu tabeli proba ustawia kolor czerwony.
void sprawdz(tmp *data)	Funkcja sprawdza czy wszystkie elementy z pojedynczego wiersza tabeli proba posiadają już kolor.
void sprawdzanie(GtkWidget *widget, tmp* data)	Funkcja dla trybu 1vs1 sprawdza poprawność kul, generując czarną kule gdy kolor i położenie się zgadza lub białą gdy kolor się zgadza ale pozycja już nie. Funkcja również generuje komunika o rezultacie w odgadywaniu, rozpoczyna 2 rundę oraz podsumowuje grę komunikatem kto wygrał rozgrywkę.
void sprawdzaniee(GtkWidget *widget, tmp* data)	Funkcja dla trybu Odgadnij kod sprawdza poprawność kul, generując czarną kule gdy kolor i położenie się zgadza lub białą gdy kolor się zgadza ale pozycja już nie. Funkcja również generuje komunika o rezultacie w odgadywaniu.

## **5. Testowanie**

Program został przetestowany dla różnych kombinacji kolorów(4 kolory takie same, wszystkie różne, itp.). Za każdym razem kule informujące o poprawność były generowane w sposób prawidłowy

## **6. Wnioski**

Program został wykonany i działa poprawnie.