

Podstawy programowania, Czwartek 21.05.2015, 16:15-17:45

Projekt, część 1

1. Zadanie

Projekt polega na stworzeniu logicznej gry komputerowej działającej w trybie tekstowym o nazwie „Minefield”.



2. Cele

Celem projektu jest napisanie bardziej rozbudowanego programu niż było to możliwe na pojedynczych zajęciach i przeciwiczenie wcześniej nauczonych mechanizmów języka C. Ponieważ jest to projekt, wymagana jest również własna kreatywność oraz inwencja – nie wszystko w treści zadania będzie napisane krok po kroku jak to zrobić.

Uwaga! Zalecane jest szerokie stosowanie zewnętrznych funkcji – w dobrym projekcie funkcja `main` powinna być jak najkrótsza.

3. Menu główne

Od razu po uruchomieniu program powinien wyświetlić menu główne, w którym znajdować się będzie opis możliwych do wyboru opcji. Menu główne powinno wyglądać mniej więcej tak:

```
wfpw@meyrin: ~/Dropbox/Nasze_pPb/Uczelnia/PP/2015/lgraczyk/proj
WITAJ W GRZE... MINEFIELD!!!!
Co chcesz zrobic?
1. Nowa gra
2. Najlepsze wyniki
3. Informacje o grze i autorze
0. Wyjście
Wybieram: █
```

W części 1 projektu obsłużymy opcje „1”, „3” i „0” (wpisanie „2” powinno ponownie wyświetlić menu).

Menu działa w nieskończonej pętli, której przerwanie i wyjście z programu następuje po wpisaniu opcji „0”. Wybór każdej z opcji jest dokonywany poprzez instrukcję `switch-case`.

3. Informacja o autorze

Wybranie opcji „3” powinno wyświetlić na ekranie informację o autorze a następnie wrócić do menu głównego:

```
wfpw@meyrin: ~/Dropbox/Nasze_pPb/Uczelnia/PP/2015/lgraczyk/proj
WITAJ W GRZE... MINEFIELD!!!!
Co chcesz zrobic?
1. Nowa gra
2. Najlepsze wyniki
3. Informacje o grze i autorze
0. Wyjscie
Wybieram: 3
-----MINEFIELD!-----
Gra napisana przez: Imie Nazwisko
Projekt z Podstaw Programowania 2015

WITAJ W GRZE... MINEFIELD!!!!
Co chcesz zrobic?
1. Nowa gra
2. Najlepsze wyniki
3. Informacje o grze i autorze
0. Wyjscie
Wybieram: █
```

4. Nowa gra

To będzie najważniejsza część przyszłej gry, dlatego należy ją obowiązkowo napisać w osobnej funkcji. Wybranie opcji „1” w menu głównym powinno wyświetlić na ekranie komunikat o rozpoczęciu gry, poprosić użytkownika o podanie liczby min, które zostaną ustawione w grze (zostaną wylosowane ich współrzędne), oraz wyświetlić ustawienie min na przyszłej planszy (implementacja – patrz poniżej):

```
wfpw@meyrin: ~/Dropbox/Nasze_pPb/Uczelnia/PP/2015/lgraczyk/proj
WITAJ W GRZE... MINEFIELD!!!!
Co chcesz zrobic?
1. Nowa gra
2. Najlepsze wyniki
3. Informacje o grze i autorze
0. Wyjscie
Wybieram: 1
-----MINEFIELD!-----
Rozpoczynamy nowa gre
Podaj liczbe min:
10
Mina nr: 0, x: 29, y: 12
Mina nr: 1, x: 24, y: 11
Mina nr: 2, x: 17, y: 21
Mina nr: 3, x: 2, y: 29
Mina nr: 4, x: 30, y: 23
Mina nr: 5, x: 21, y: 33
Mina nr: 6, x: 19, y: 33
Mina nr: 7, x: 14, y: 14
Mina nr: 8, x: 17, y: 27
Mina nr: 9, x: 18, y: 16
█
```

Szczegóły implementacji

W programie powinna znaleźć się struktura `Punkt`, która posiada dwa parametry: `int x, int y`. Strukturę obsługuje funkcja `void UstawPunkt(Punkt *p, int X, int Y)`.

W programie należy zaimplementować drugą strukturę `PoleMinowe`, która posiada dwa parametry: `int liczbaMin, Punkt *miny`. W strukturze mamy jednowymiarową tablicę typu `Punkt` o nazwie `miny`. Strukturę obsługuje funkcja `void UstawMiny(PoleMinowe *p, int LiczbaMin)`, która dla danego obiektu typu `PoleMinowe` ustawia liczbę min oraz alokuje pamięć tablicy `miny` na rozmiar `LiczbaMin`. Po zaalokowaniu pamięci tablicy, funkcja ustawia współrzędne `x, y` (losowo, od wartości 1 do 39) każdej miny.

Struktury należy umieścić w pliku `.h` (można obie umieścić w 1 pliku), zaś funkcje obsługujące struktury w pliku `.c` (również wystarczy 1 plik).

Uwaga!

Bardzo użyteczna komenda:

```
system("clear");
```

z biblioteki **stdlib.h**

na systemach UNIX'owych czyści całe okno terminala: umożliwia to np. zastępowanie jednego menu innym.

Punktacja:

1. Stworzenie menu głównego (**1 p.**)
2. Obsłużenie opcji wyjścia i informacji o autorze (**1 p.**)
3. Obsłużenie opcji „Nowa gra” wraz ze stworzeniem struktur i wylosowaniem położenia min (**6 p.**)

Całość kompilujemy w linii poleceń używając pliku `Makefile` oraz polecenia `make`. (**1 p.**)

Zrozumienie, analiza treści, projekt programu + poprawność i estetyka kodu. Punkty za ten element można dostać mając skończoną przynajmniej połowę 1 części programu (**1 p.**)