

Projekt: Projekt polega na napisaniu tekstowej gry komputerowej w okienku terminala

Część 1 – Kreator postaci oraz poruszanie się po planszy

Za niekompilujący się pod koniec program można uzyskać co najwyżej 4 punkty.

Oceniane są jedynie w pełni działające i przetestowane elementy programu – odpowiednie informacje powinny być wyświetlane na ekranie.

1. Struktura Bohater (1 p)

Struktura Bohater zawiera trzy pola: **x**, **y** (liczby całkowite przechowujące współrzędne bohatera na planszy) oraz **znak** (typu 'char', przechowujący ikonę oznaczającą bohatera).

2. Kreator postaci (2 p) – Tworzenie obiektu struktury Bohater

- Kreator postaci jest uruchamiany na samym początku programu.

- Kreator postaci powinien się znajdować w całości w zewnętrznej funkcji, np.

```
void KreatorPostaci(Bohater *b)
```

- Kreator powinien inicjalizować obiekt struktury Bohater.

+ W funkcji `KreatorPostaci` należy ustawić domyślne wartości dla utworzonego obiektu:

```
x = 10, y = 10, znak = podany z klawiatury.
```

3. Rozgrywka – osobna funkcja

Jest to funkcja która będzie przechowywała wszystkie zdarzenia które będą się przydawać naszemu bohaterowi. Podczas tej części projektu ma za zadanie narysować planszę i umożliwiać ruch bohatera (utworzonego przy użyciu Kreatora) po planszy. Rozgrywkę należy rozpocząć od wyczyszczenia ekranu z menu przy użyciu komendy `system("clear")` z biblioteki `stdlib.h`.

3a. Narysowanie planszy (3.5 p):

W celu jej stworzenia najprościej napisać funkcję

```
void plansza(Bohater b)
```

wywoływaną w funkcji `Rozgrywka`, która przyjmuje za argument bohatera. Zakładamy, że plansza jest stałego rozmiaru 20x20 (wartość tą można zdefiniować poprzez stałą globalną `#define PLANSZA 20`). Przedstawiona po prawej plansza (z ramką!) jest wynikiem zastosowania odpowiedniej kombinacji funkcji `printf` i instrukcji warunkowych `if` umieszczonych wewnątrz dwóch zagnieżdżonych pętli `for`.

Dla każdego punktu (x,y) planszy musimy sprawdzić:

- czy odpowiada on współrzędnym położenia bohatera:
 - jeśli dany punkt planszy nie odpowiada położeniu bohatera wypisujemy na ekran spację,
 - jeśli zaś odpowiada, to wypisujemy znak [odczytany ze struktury] bohatera w danym punkcie.

```
=====
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
I             I
=====
[w,s,a,d] - ruch bohatera
[0] - wyjście
```

Bohatera domyślnie (na początku gry) umieszczamy w punkcie (x,y) zapisanym w składnikach obiektu Bohater b.

3b. Ruch bohatera (3.5 p):

Ruch bohatera polega na zmianie położenia (czyli wartości x oraz y) bohatera b oraz przerysowywaniu planszy po każdej takiej zmianie. W tym celu należy wewnątrz funkcji `Rozgrywka` zaimplementować nieskończoną pętlę, w której każda zmiana pozycji będzie powodować czyszczenie ekranu terminala (znów komenda `system("clear")`) oraz ponowne narysowanie planszy poprzez wywołanie funkcji `plansza`.

Jedyną niewiadomą pozostaje zmiana pozycji bohatera – jak to zrobić? Do tego celu należy wykorzystać funkcję `char getch()` z pliku `getchar.h` (jest to podobna funkcja do `scanf`, z tą różnicą, że nie wymaga potwierdzenia wprowadzenia znaku poprzez wciśnięcie klawisza [Enter]. Ewentualnie można użyć również popularnego „scanf’a”, jednakże proszę wtedy pamiętać o potwierdzaniu enterem każdego ruchu).

Więcej informacji na następnej stronie.

Przykład wykorzystania funkcji `char getch()` - wczytanie pojedynczego znaku z klawiatury i jego wypisanie:

```
#include <stdio.h>
#include "getchar.h"

int main(void)
{
    char znak;
    znak = getch();
    printf("Podany znak to: %c\n", znak);

    return 1;
}
```

Uwaga! Żeby kod zadziałał należy również ściągnąć plik `getchar.h` ze strony oraz umieścić w katalogu programu!

Co należy zrobić z wczytanym znakiem? Wciśnięcie klawiszy:

- „w” - przesuwa bohatera o 1 w górę (czyli wartość y położenia bohatera zmniejsza się o 1)
- „s” - przesuwa bohatera o 1 w dół (czyli wartość y położenia bohatera zwiększa się o 1)
- „a” - przesuwa bohatera o 1 w lewo (czyli wartość x położenia bohatera zmniejsza się o 1)
- „d” - przesuwa bohatera o 1 w prawo (czyli wartość x położenia bohatera zwiększa się o 1)
- „0” - wychodzi z gry