

POPRO – lab 6

Tablice, wskaźniki do tablic, struktury

Autor: Sebastian Kozłowski, Daniel Paczesny, Krystian Ignasiak

1 Tematyka

Laboratorium 6 z przedmiotu POPRO dotyczy zagadnień związanych z tablicami, w tym tablicami wielowymiarowymi, wskaźnikami do tablic, strukturami *struct* oraz możliwościami rysowania na standardowym wyjściu.

2 Przygotowanie do zajęć laboratoryjnych

W ramach przygotowania do laboratorium warto zapoznać się z materiałami z zakresu tworzenia tablic wielowymiarowych, wskaźników do tablic oraz struktur.

1. <https://beginnersbook.com/2014/01/2d-arrays-in-c-example/>
2. <https://beginnersbook.com/2014/01/c-passing-array-to-function-example/>
3. <https://beginnersbook.com/2014/01/c-structures-examples/>

3 Sprawozdanie

Sprawozdanie stanowią pliki *.c* z rozwiązaniami zadań, po jednym dla każdego zadania. Pliki powinny zostać umieszczone w repozytorium. Odpowiedzi na pytania powinny przyjmować formę komentarzy umieszczonych w odpowiednich miejscach kodu.

Uwaga! W każdym pliku stanowiącym podstawę do wystawienia punktów pierwszą linię powinien stanowić komentarz zawierający imię i nazwisko autora, numer grupy laboratoryjnej (LAB101/LAB102) oraz numer zadania.

4 Przebieg laboratorium

W trakcie laboratorium należy napisać sprawdzian wejściowy (wejściówka) wg wskazówek prowadzącego oraz wykonać trzy zadania. Wejściówka, zadanie pierwsze i drugie warte są 1 punkt każde, zadanie trzecie 2 punkty.

4.1 Zadanie 1 – Tablice i wskaźniki do tablic (1 pkt.)

Napisz program, który deklaruje tablicę wartości typu *int* o wymiarach 5 na 10 i przypisuje jej elementom losowe wartości początkowe. Zakres wartości początkowych jest podany przez prowadzącego laboratorium w trakcie zajęć. Program następnie wyświetla wylosowane wartości w czytelnej formie. Zadeklarowaną tablicę należy potraktować jako 5 wektorów, z których każdy zawiera 10 elementów. Wyznacz następujące normy dla każdego z wektorów:

- norma typu *maksimum*,
- norma euklidesowa.

Wyznacz wiersz, dla którego:

- norma typu *maksimum* przyjmuje wartość maksymalną,
- norma euklidesowa przyjmuje wartość maksymalną.

Wyświetlaniem, obliczaniem norm, obliczaniem numerów wierszy powinny zajmować się osobne funkcje. Wyjaśnij w komentarzu:

- dlaczego do funkcji przetwarzającej dane tablicowe przekazywany jest rozmiar tablicy,
- dlaczego w argumencie formalnym funkcji reprezentującym tablicę możemy użyć notacji tablicowej lub notacji wskaźnikowej.

4.2 Zadanie 2 – Struktury i typedef (1 pkt.)

Dana jest następująca struktura:

```
#include <stdio.h>

struct Fuel {
    float distance;
    float volume;
    float performance;      /* the distance possible to cover using a unit volume of fuel */
};

typedef struct Fuel Fuel;

int main(void) {
    //testowanie utworzonych funkcji
}

/* implementacja funkcji obliczająca wartość zmiennej performance
   funkcja pobiera strukturę Fuel
   funkcja zwraca kompletną strukturę
```

Projekt „NERW 2 PW. Nauka – Edukacja – Rozwój – Współpraca” współfinansowany jest ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego



Fundusze Europejskie
Wiedza Edukacja Rozwój



Rzeczpospolita
Polska

Politechnika
Warszawska

Unia Europejska
Europejski Fundusz Społeczny



*/

```
/* implementacja funkcji obliczającej wartość zmiennej performance
   funkcja pobiera adres struktury Fuel
   funkcja modyfikuje pole przekazanej struktury
*/
```

- Opracuj funkcję, która pobiera argument typu *struct Fuel*, oblicza wartość pola *performance* (liczba kilometrów, jakie można przejechać na jednym litrze paliwa) i zwraca kompletną strukturę. Należy przyjąć, że przekazana struktura zawiera wartości pól *distance* i *volume*, przy czym odległość jest w kilometrach, objętość w litrach. Napisz kod, który będzie sprawdzał dane wejściowe.
- Opracuj funkcję, która pobiera adres argumentu typu *struct Fuel*, oblicza wartość pola *performance* (liczbę kilometrów, jakie można przejechać na jednym litrze paliwa) i przypisuje ją odpowiedniemu polu.
- Opracuj funkcję, która pobiera adres argumentu typu *struct Fuel*, po czym przelicza wartość pola *distance* z kilometrów na mile (1 mila = 1.6 km), przelicza wartość pola *volume* z litrów na galony (1 galon = 3.8 litra), po czym polu *performance* przypisuje liczbę mil możliwych do przejechania na jednym galonie paliwa.

4.3 Zadanie 3 – Rysowanie na standardowym wyjściu (2 pkt.)

Dana jest struktura:

```
struct Punkt {
    double x;
    double y;
};

typedef struct Punkt Punkt;
```

Napisz program, w którym zostanie zadeklarowana i zdefiniowana 20-elementowa tablica struktur typu *struct Punkt*. Struktura *Punkt* reprezentuje punkt w układzie współrzędnych kartezjańskich. Wypełnij poszczególne pola struktur *Punkt* tak, aby wartość pola *x* zmieniała się w sposób dyskretny od 0 do 19, natomiast wartość pola *y* wynosiła $y = 0.5x$. Napisz funkcję, która będzie wyświetlała otrzymany wykres w oknie terminala. Zabezpiecz tablicę struktur przed modyfikacją wewnątrz funkcji.