Politechnika Wrocławska

Dorota Aniszewska, Wojciech Myszka

PROGRAMOWANIE W JĘZYKU C
Lista nr 6. Tablice dwuwymiarowe. Pliki.

Przekazywanie tablic dwuwymiarowych do funkcji zawarto w programie tablice2D. Informacje na temat dynamicznego przydzielania pamięci są opisane na stronie tab2d.html.
Przykładowe programy znajdują się w plikach pliki1.c, pliki2.c, pliki3.c.

1. Napisz program, który wyznacza macierz transponowaną. Tablica dwuwymiarowa może mieć rozmiary określone przez stałe, wartości elementów należy wylosować.

2. Napisz funkcje, które wykonują podstawowe operacje arytmetyczne dla dwóch macierzy, czyli tablic dwuwymiarowych. Utwórz plik tekstowy (plik umieść w tym samym katalogu) i za pomocą edytora tekstu zapisz w nim wymiary tych macierzy. Po odczytaniu wymiarów z pliku, jeśli wykonanie operacji arytmetycznych jest możliwe, program powinien przydzielić odpowiednią ilość pamięci dla każdej tablicy oraz tablic mających zawierać sumę, różnicę, iloczyn macierzy oraz wykonać obliczenia.

3. Napisz funkcję, który oblicza wyznacznik macierzy. Tablica dwuwymiarowa może mieć rozmiar losowany z pewnego zakresu. Funkcja ta powinna być rekurencyjna.

4. Napisz program, który zapisze do pliku tekstowego dane (nazwisko, imię, wiek) przynajmniej 20 osób. Można to wykonać deklarując tablicę nazwisk oraz wypełniając ją (np. za pomocą instrukcji *char nazwiska[][20]={ "Nowak", "Rojek", "Walc", "Bikol" };*) oraz tablicę imion, po czym do każdej kombinacji nazwiska i imienia zapisywanego do pliku wylosować wiek. W pliku tekstowym dane każdej osoby powinny być w osobnej linijce.

5. Napisz program czytający z pliku informacje i wyświetlający każdy bajt w kilku formatach. Na przykład tak: plik tekstowy zawierający takie znaki (będące jednocześnie programem w C):

```
#include <math.h>
#include <stdio.h>
int main()
{
   printf( "%lf\n", pow(2., 3.) );
   return 0;
}
```

ma być wyświetlony następująco:

```
0000000 # i n c l u d e < m a t h . h
        35 105 110 99 108 117 100 101 32 60 109 97 116 104 46 104
        23 69 6e 63 6c 75 64 65 20 3c 6d 61 74 68 2e 68
        043 151 156 143 154 165 144 145 040 074 155 141 164 150 056 150
0000016 > \n # i n c l u d e < s t d i
        62 10 35 105 110 99 108 117 100 101 32 60 115 116 100 105
        3e 0a 23 69 6e 63 6c 75 64 65 20 3c 73 74 64 69
        076 012 043 151 156 143 154 165 144 145 040 074 163 164 144 151
```

```
0000032 o . h > \n i n t m a i n ( ) \n
        111 46 104 62 10 105 110 116 32 109 97 105 110 40 41 10
        6f 2e 68 3e 0a 69 6e 74 20 6d 61 69 6e 28 29 0a
        157 056 150 076 012 151 156 164 040 155 141 151 156 050 051 012
0000048 { \n p r i n t f ( " %
        123 10 32 32 32 32 112 114 105 110 116 102 40 32 34 37
        7b 0a 20 20 20 20 70 72 69 6e 74 66 28 20 22 25
        173 012 040 040 040 040 160 162 151 156 164 146 050 040 042 045
0000064 l f \ n " , p o w ( 2 . , 3
        108 102 92 110 34 44 32 112 111 119 40 50 46 44 32 51
        6c 66 5c 6e 22 2c 20 70 6f 77 28 32 2e 2c 20 33
        154 146 134 156 042 054 040 160 157 167 050 062 056 054 040 063
0000080 . ) ) ; \n r e t u r n
        46 41 32 41 59 10 32 32 32 32 114 101 116 117 114 110
        2e 29 20 29 3b 0a 20 20 20 20 72 65 74 75 72 6e
        056 051 040 051 073 012 040 040 040 040 162 145 164 165 162 156
0000096 0 ; \n } \n
        32 48 59 10 125 10
        20 30 3b 0a 7d 0a
        040 060 073 012 175 012
0000102
```

Najpierw jest numer bajtu od początku pliku. Później zawartość tekstowa. Kolejne trzy linie zawierają zawartość każdego baju wyświetlaną dziesiętnie, szesnastkowo i ósemkowo.