

APLICAREA MĂRIMILOR TABELARE NEREGULATE ÎN PROGRAMARE

Vitalie ȚÎCĂU, lector universitar

Universitatea de Stat „Alec Russo” din Bălți

Rezumat. Conceptul de tip de date este de bază într-un limbaj de programare. Astfel, este actuală utilizarea materiale didactice ce permit gestiunea tipurilor, în special a tablourilor bidimensionale. Mărimile tabelare sunt destinate pentru reprezentarea datelor omogene. Deseori se aplică tablouri rectangulare (regulate). Însă în practică se utilizează și tabele neregulate, când numărul de elemente de pe linii este diferit, tipul datelor fiind același. Tablourile neregulate nu sunt abordate de orice limbaj de programare și sunt restrâns descrise. În lucrare este cercetată abordarea aplicării tabelor bidimensionale neregulate în limbajul Visual C# cu aplicații în practică.

Abstract. The concept of data type is basic in a programming language. Thus, it is current to use teaching materials that allow the management of types, especially two-dimensional arrays. Arrays used for the representation of homogeneous data. Rectangular (regular) arrays are often applied. In practice, jagged array also used, when the number of elements on the lines is different, their type being the same. Jagged array do not described by any programming language and are narrowly described. The paper researches the approach of the applications of two-dimensional jagged array in the Visual C # language with applications in practice.

Cuvinte-cheie: tablouri neregulate, programare C#, aplicații în practică.

Keywords: jagged tables, C # programming, applications in practice.

Introducere

În lucrare sunt cercetate modurile de descriere, alocare a memoriei, inițializare și aplicare a tabelor bidimensionale regulate și neregulate. Sunt definiți algoritmi de prelucrare a tablourilor regulate și neregulate în Visual C#. Este cercetat modul de rezolvare a unor probleme, bazate pe aplicarea tablourilor bidimensionale regulate și neregulate în limbajul C#:

- gestiunea unui tablou abstract de numere întregi;
- prelucrarea notelor elevilor (studenților);
- gestiunea figurilor din plan;
- determinarea profitului la realizarea mărfurilor unor companii.

Metode și materiale aplicate

În C# există două categorii de tipuri de date: *tipuri valoare* și *tipuri referință*. Toate tipurile valoare sunt derivate din clasa *System.ValueType*, care la rândul ei este derivată din clasa *Object*. Nu este posibil ca dintr-un tip valoare să se deriveze [1]. O variabilă de tip valoare întotdeauna trebuie să conțină o valoare (nu poate fi *null*, spre deosebire de tipurile referință).

Tabloul reprezintă un șir ordonat de elemente de același tip. Tabloul se poate caracteriza prin nume, dimensiune, numărul de elemente. Tablourilor li se atribuie aceleași tipuri de date și clase de memorie, ca și variabilelor simple.

În cele mai multe cazuri nu se cunoaște aprioric numărul de obiecte care se vor folosi la momentul rulării. O soluție pentru stocarea elementelor având același tip o reprezintă tablourile. Sintaxa de declarare și utilizare este asemănătoare în C# cu cea din Java sau C++, dar fiecare tablou este un obiect, derivat din clasa abstractă *System.Array*. O variabilă de tip tablou poate avea valoare de *null* sau poate să indice către o instanță validă. Pentru a putea reține date în structura de tip tablou, este necesară o operație de instanțiere.

În cazul tablourilor cu mai multe dimensiuni se face distincție între tablouri regulate și tablouri neregulate (tablouri de tablouri).

Matricea multidimensională are două sau mai multe dimensiuni, un element al acestora se poate accesa prin combinarea a doi sau mai mulți indici. *Tablourile rectangulare* au proprietatea că pe fiecare dimensiune se află același număr de elemente [2]. Fiecare element este specificat cu doi indici. Un indice indică rândurile și un indice indică coloanele. Pentru a declara două dimensiuni se aplică virgula în prima paranteză pătrată.

Un tablou neregulat (*jagged arrays*) reprezintă un tablou de tablouri [2]. Deci, lungimea rândurilor nu este aceeași. Declararea unui tablou neregulat cu două dimensiuni se face folosind seturi de paranteze pătrate pentru a indica fiecare dimensiune. Pentru accesul la un element se înscriu doi indici în paranteze pătrate aparte. Exemplu de matrice eșalonată cu două dimensiuni:

```
int[][] jagged = new int[2][];    jagged[0] = new int[4]; jagged[1] = new int[3];
```

Pentru a ști numărul de elemente ale tabloului putem folosi *array.Length*. Folosind proprietatea *Length* pentru matrice eșalonate se obține lungimea pentru fiecare tablou-rând.

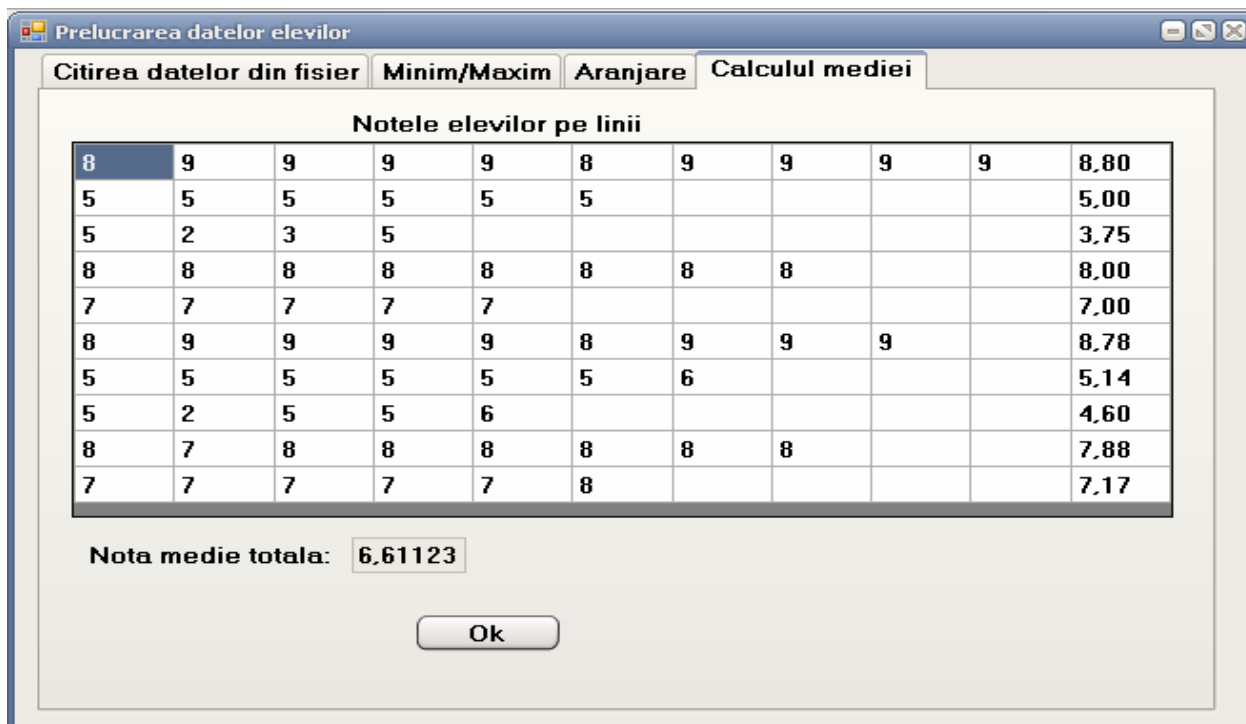


Figura 1. Calculul mediei notelor elevilor

Notele elevilor (studentilor). Se cunosc notele elevilor la un obiect, obținute pe parcursul unui trimestru. Evident, că numărul de note nu este omogen: unii elevi (studenți) au mai multe note, iar alții – mai puține. Nota medie se calculează în baza notelor obținute. Datele inițiale se păstrează într-un fișier text, în care pe primul rând se înscrie numărul de elevi, iar pe următoarele rânduri se înscrie numărul de note și, respectiv, notele. În figura 1 este prezentat exemplul de calcul al mediei elevilor (studentilor).

Determinarea profitului la realizarea mărfurilor unei companii. Fie se cunosc profiturile unei companii la realizarea mărfurilor. Numărul de mărfuri diferă de la o companie la alta. De determinat profitul total și rata profitului companiilor.

Valorile reale ale profitului se citesc din fișier pe rânduri ca șiruri de caractere, se despart folosind metoda *Split* și se convertesc în numere reale. Cu ajutorul metodelor standard *Max()* și *Min()* se determină care produs și a cărei firme aduce profitul maximal, respectiv, minimal.

De asemenea se aplică metodele standard *Sort()* și *Reverse()* ale clasei *Array* pentru aranjare crescătoare, respectiv, descrescătoare a profiturilor fiecărei firme în parte.

Gestiunea figurilor. Pentru figurile din spațiu de forma: cerc, triunghi, pătrat, dreptunghi, romb, paralelogram și trapez se cunosc caracteristicile lor: lungimile laturilor și mărimea unghiului de bază. Se ce re de determinat aria figurilor și perimetrul (sau lungimea pentru cerc).

Numele figurilor și caracteristicile se citesc din fișier pe rânduri ca șiruri de caractere.

În figura 2 este prezentat exemplu de calcul al ariei figurilor.

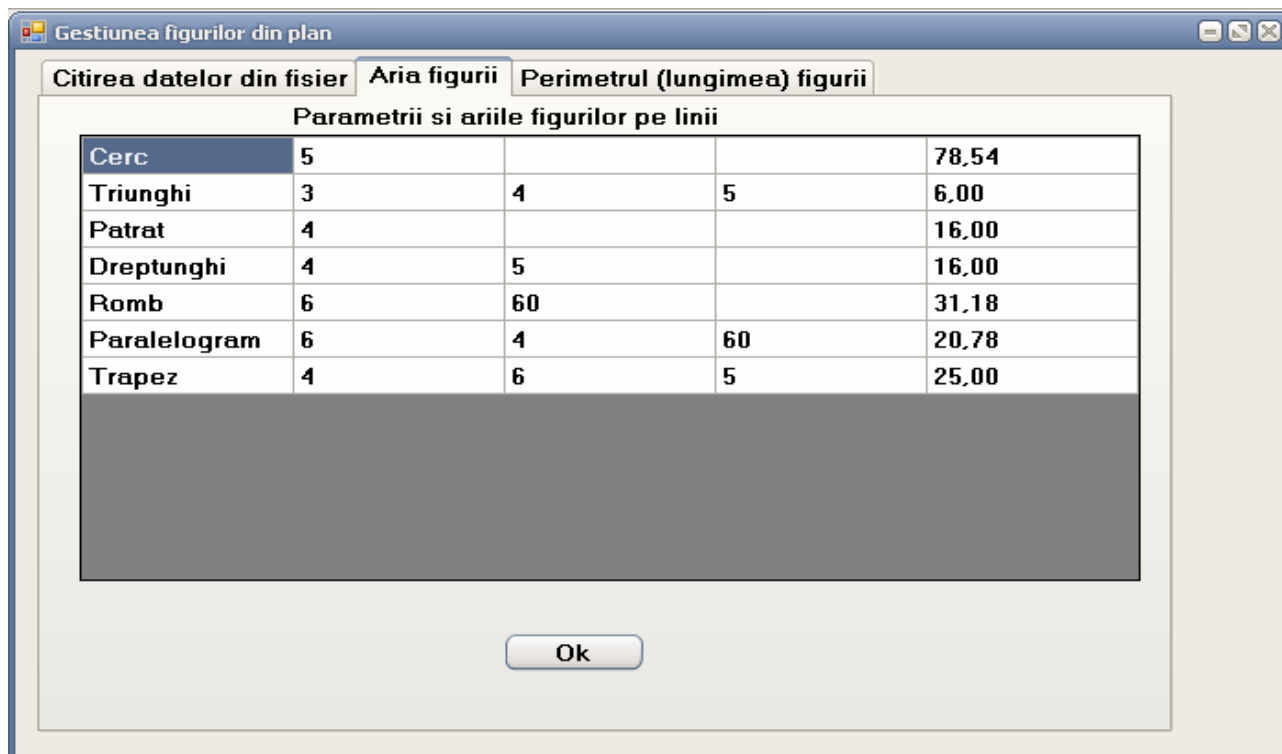


Figura 2. Calculul ariei figurilor

Concluzii

În lucrare sunt descrise tablourile bidimensionale regulate și neregulate. În Visual C# pe platforma Microsoft Visual Studio a fost elaborată o aplicație, care rezolvă un set de probleme cu aplicarea tablourilor bidimensionale neregulate. Programul permite utilizarea exemplelor tipice.

Aplicarea tablourilor neregulate este utilă în practică, deoarece propune un instrument folositor de prelucrare a datelor. Legătura strânsă a tablourilor unidimensionale de cele bidimensionale neregulate face ca algoritmi de prelucrare a tablourilor bidimensionale neregulate să se bazeze pe algoritmi de prelucrare a tablourilor unidimensionale.

Bibliografie

1. PELLAND, Patrice. *Build a Program. NOW, Microsoft Visual C# 2005*. Express Edition, eBook, Microsoft Press, 2005. 224 p. ISBN: 0-7356-2229-9.
2. ПАВЛОВСКАЯ, Татьяна. *C#. Программирование на языке высокого уровня*. СПб.: Питер, 2021, 432 с. ISBN: 978-5-4461-0913-5.