

Інститут ІКНІ

Кафедра ПЗ

ЗВІТ

До лабораторної роботи № 3

На тему: “Робота з масивами в C”

З дисципліни: “Основи програмування”

Лектор:

ст.викл. каф. ПЗ

Муха Т.О.

Виконав:

ст. гр. ПЗ-16

Шевчук С.А

Прийняла:

асист. каф. ПЗ

Бутрак І. О.

« _____ » _____ 2021 р.

Σ = _____.

Тема: робота з масивами в С.

Мета роботи: навчитися працювати з масивами в С.

ЛАБОРАТОРНЕ ЗАВДАННЯ

1. Ознайомитися з теоретичним матеріалом викладеним вище в даній інструкції і виконати приклади програм.
2. Одержати індивідуальне завдання з Додатку 1.
3. Розробити алгоритм розв'язання індивідуального завдання і подати його у вигляді блок схеми.
4. Скласти програму на мові С у відповідності з розробленим алгоритмом.
5. Виконати обчислення по програмі.
6. Одержати індивідуальне завдання з Додатку 2.
7. Розробити алгоритм розв'язання індивідуального завдання і подати його у вигляді блок схеми.
8. Скласти програму на мові С у відповідності з розробленим алгоритмом.
9. Виконати обчислення по програмі при різних значеннях точності і порівняти отримані результати.
10. Підготувати та здати звіт про виконання лабораторної роботи.

ІНДИВІДУАЛЬНІ ЗАВДАННЯ

1. Написати програму для обробки даних, організованих у масив, згідно завдання наведеного варіанту. У програмі використати такі форми звертання до елементів масиву:
 - на основі індексів,
 - через розадресоване ім'я масиву і зміщення елемента.

Ввести вектор дійсних чисел. Всі елементи вектора, що розмішені перед його мінімальним елементом, зменшити на половину від середнього арифметичного елементів введеного вектора.

2. Написати програму для обробки даних, організованих у масив, згідно завдання наведеного варіанту. У програмах використати різні форми звертання до елементів багатовимірних масивів, у тому числі за допомогою конструкцій “вказівник на масив” і “вказівник на вказівник”. Враховувати, що кожен рядок матриці можна опрацьовувати як окремий елемент.

int A[15][15], B[15][15]; Знайти C=AB.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Масив являє собою сукупність однотипних змінних, розміщених у послідовно пронумерованих суміжних комірках пам'яті. Номер елемента масиву задається індексом. Індексація елементів масиву в С починається з 0. Якщо у масиві N елементів, то перший елемент матиме індекс 0, а останній – індекс (N–1).

За способом зв'язування індексів з комірками пам'яті виділяють три категорії масивів: статичні, фіксовані автоматичні та динамічні.

Статичним називають масив, в якому зв'язування індексів та розміщення в пам'яті виконуються на етапі компіляції програми. Статичні масиви досить ефективні, оскільки для їх створення та знищення не потрібно додаткових операцій.

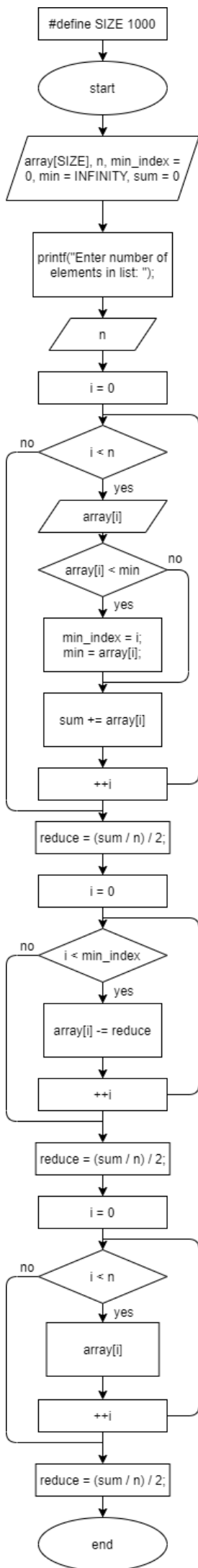
Фіксованим автоматичним називають масив, в якому індекси зв'язуються статично, а розміщення в пам'яті виконується при обробці оголошень всередині функцій.

Динамічним називають масив, де зв'язування індексів і розміщення в пам'яті виконується безпосередньо під час виконання програми. Цей спосіб організації даних достатньо гнучкий, оскільки розміри динамічного масиву можуть збільшуватися та зменшуватися під час виконання програми у міру необхідності.

Об'єм пам'яті, який займає масив, рівний добутку розміру типу елементів масиву та кількості елементів в масиві. Наприклад, якщо масив містить 16 чисел типу `int` і тип `int` займає в пам'яті 2 байти, то цілий масив займатиме $2 \times 16 = 32$ байти.

У мові C не перевіряється вихід індексу масиву за межі допустимого значення. Вихід за межі пам'яті, яку займає масив, не контролюється. При помилці програміста може відбутися запис інформації в сусідні комірки пам'яті, а наслідки цього непередбачувані. Масиви можуть бути одновимірними та багатовимірними.

БЛОК-СХЕМА №1



Мал. 1. Блок-схема програми № 1

КОД ПРОГРАМИ №1

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#define SIZE 1000

int main() {
    double array[SIZE];
    int n, min_index = 0;
    double min = INFINITY, sum = 0;

    printf("Enter number of elements in list: ");
    scanf_s("%d", &n);

    for (int i = 0; i < n; ++i) {
        scanf_s("%lf", array + i);
        if (array[i] < min) {
            min_index = i;
            min = array[i];
        }
        sum += array[i];
    }

    double reduce = (sum / n) / 2;

    for (int i = 0; i < min_index; ++i) {
        array[i] -= reduce;
    }

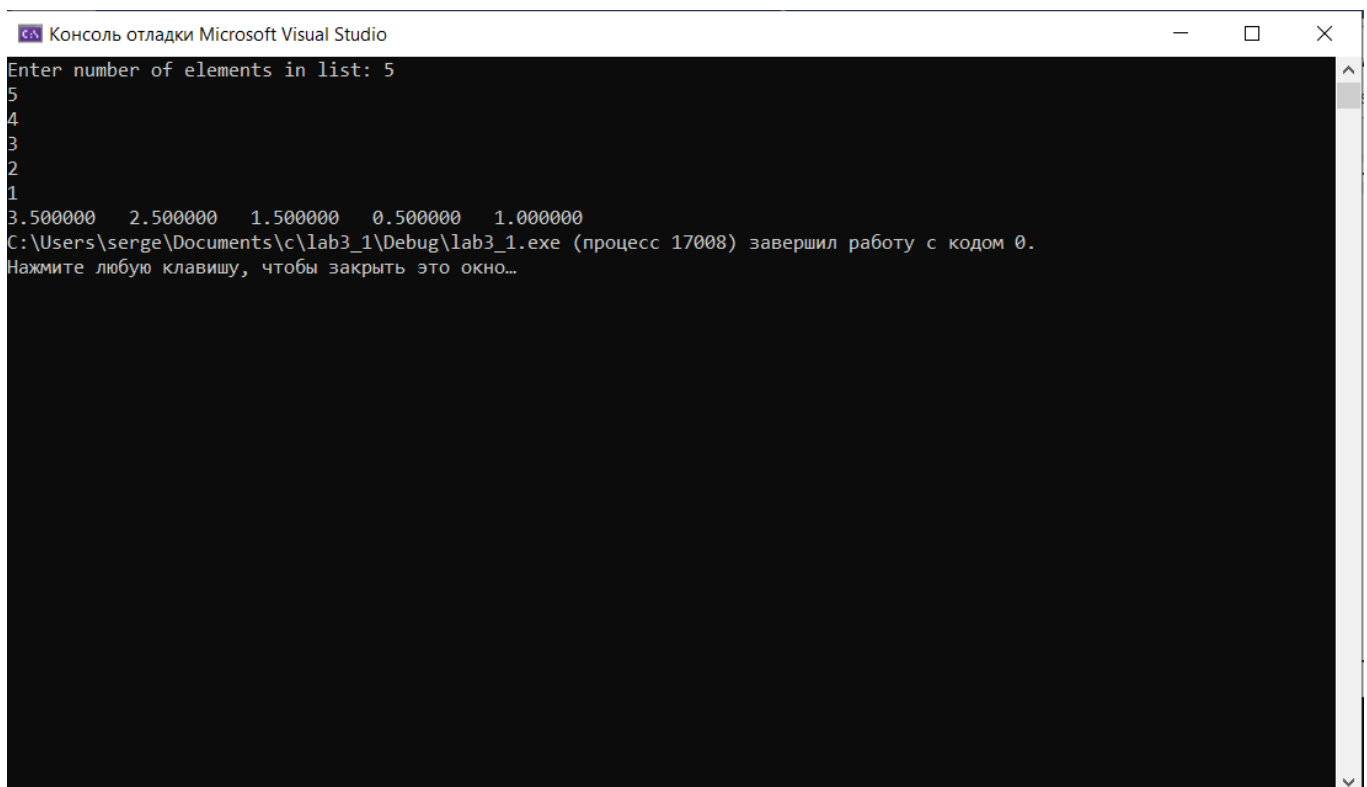
    for (int i = 0; i < n; ++i) {
        printf("%lf ", array[i]);
    }
    return 0;
}
```

ПРОТОКОЛ РОБОТИ №1

Протокол вводу: 5

5 4 3 2 1;

Протокол виводу: 3.500000 2.500000 1.500000 0.500000 1.000000



```
Консоль отладки Microsoft Visual Studio
Enter number of elements in list: 5
5
4
3
2
1
3.500000 2.500000 1.500000 0.500000 1.000000
C:\Users\serge\Documents\c\lab3_1\Debug\lab3_1.exe (процесс 17008) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...
```

Мал. 2. Результат виконання програми №1

```

graph TD
    Start([start]) --> Init[/#define SIZE 15/]
    Init --> Read[/a[SIZE][SIZE], b[SIZE][SIZE], c[SIZE][SIZE]/]
    Read --> I0[/i = 0/]
    I0 --> Cond1{i < SIZE*SIZE}
    Cond1 -- yes --> Calc1["*(a + i) = rand() % 10;  
*(b + i) = rand() % 10;"]
    Calc1 --> Print1["*(a + i)"]
    Print1 --> Cond2{i((i + 1) % SIZE)}
    Cond2 -- yes --> Print2[/\n/]
    Print2 --> Inc1[++i]
    Cond2 -- no --> Inc1
    Inc1 --> Print3[/\n\n/]
    Print3 --> I1[/i = 0/]
    I1 --> Cond3{i < SIZE*SIZE}
    Cond3 -- yes --> Calc2["*(a + i)"]
    Calc2 --> Cond4{i((i + 1) % SIZE)}
    Cond4 -- yes --> Print4[/\n/]
    Print4 --> Inc2[++i]
    Cond4 -- no --> Inc2
    Inc2 --> Read2[/temp, len/]
    Read2 --> I2[/i = 0/]
    I2 --> Cond5{i < SIZE}
    Cond5 -- yes --> J0[/j = 0/]
    J0 --> Cond6{j < SIZE}
    Cond6 -- yes --> Read3[/temp = 0/]
    Read3 --> Read4[/n = 0/]
    Read4 --> Cond7{n < SIZE}
    Cond7 -- yes --> Calc3["temp += a[i][n] * b[n][i]"]
    Calc3 --> Inc3[++n]
    Inc3 --> Cond7
    Cond7 -- no --> Assign["c[i][i] = temp"]
    Assign --> Print5[/c[i][i]/]
    Print5 --> Inc4[++j]
    Inc4 --> Print6[/\n/]
    Print6 --> Inc5[++i]
    Inc5 --> End([end])

```

```

graph TD
    Start([Start]) --> Cond1{i < SIZE}
    Cond1 -- no --> End([end])
    Cond1 -- yes --> InitJ[/j = 0/]
    InitJ --> Cond2{j < SIZE}
    Cond2 -- no --> End
    Cond2 -- yes --> InitTemp[/temp = 0/]
    InitTemp --> InitN[/n = 0/]
    InitN --> Cond3{n < SIZE}
    Cond3 -- no --> End
    Cond3 -- yes --> CalcTemp[ $temp += a[i][n] * b[n][j]$ ]
    CalcTemp --> IncN[ $n++$ ]
    IncN --> Cond3
    IncN --> AssignC[c[i][j] = temp]
    AssignC --> IncJ[ $j++$ ]
    IncJ --> Cond2
    IncJ --> IncI[ $i++$ ]
    IncI --> Cond1

```

КОД ПРОГРАМИ №2

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <math.h>
#define SIZE 5

int main() {
    int a[SIZE][SIZE], b[SIZE][SIZE], c[SIZE][SIZE];

    srand(time(NULL));

    for (int i = 0; i < SIZE * SIZE; ++i) {
        *(&a + i) = rand() % 10;
        *(&b + i) = rand() % 10;
        printf("%d ", *(&a + i));
        if (!(i + 1) % SIZE) printf("\n");
    }

    printf("\n\n");

    for (int i = 0; i < SIZE * SIZE; ++i) {
        printf("%d ", *(&b + i));
        if (!(i + 1) % SIZE) printf("\n");
    }

    int temp, len;
    printf("\n\n");

    for (int i = 0; i < SIZE; ++i) {
        for (int j = 0; j < SIZE; ++j) {
            temp = 0;
            for (int n = 0; n < SIZE; ++n) {
                temp += a[i][n] * b[n][j];
            }
            c[i][j] = temp;
            printf("%-4d ", c[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

ПРОТОКОЛ РОБОТИ №2

Протокол виводу:

8 8 0 9 2 3 4 4 6 8 9 2 8 7 6
8 2 4 9 3 2 0 6 8 2 0 7 9 8 6
0 8 3 1 7 8 5 2 5 6 5 0 6 4 4
1 5 4 0 7 3 2 0 1 2 4 1 3 9 9
2 0 6 0 6 0 6 2 5 3 7 4 9 8 9
4 1 0 2 6 4 0 9 8 8 0 7 6 9 5
3 5 2 0 4 1 1 6 8 8 0 1 7 4 1
7 8 0 9 2 9 7 5 8 6 0 4 6 8 1
2 4 9 1 5 7 8 5 5 5 4 2 4 8 9
5 1 4 3 2 4 9 7 4 9 2 9 2 8 2
9 2 2 1 9 9 0 6 1 2 1 1 8 1 7
6 9 0 1 3 7 0 1 0 7 5 4 6 2 3
3 6 3 3 5 2 2 4 1 5 8 5 7 4 8
5 3 1 0 1 2 4 1 7 5 5 9 8 7 8
9 0 0 2 2 0 1 7 9 6 7 9 1 3 1

4 8 8 5 2 2 9 7 2 1 5 0 6 1 7
5 8 7 9 8 1 0 0 9 3 2 5 6 9 5
5 0 3 0 6 1 3 8 6 0 3 8 4 0 2
3 3 6 3 5 3 2 0 5 4 4 3 1 5 0
7 6 5 7 1 0 1 9 0 3 4 6 7 6 5
2 1 9 7 1 2 7 5 8 8 8 0 2 4 0
2 4 0 1 6 0 4 2 4 7 8 4 2 8 3
8 1 1 3 8 2 4 2 6 8 1 0 6 5 2
8 1 4 5 9 0 5 8 4 0 1 1 3 9 3
9 5 0 1 0 0 3 8 1 7 9 2 3 4 5
0 6 3 0 2 7 3 8 4 4 2 9 8 7 9
2 7 1 1 6 2 1 5 5 7 3 9 2 3 1
9 4 9 4 8 7 3 6 0 0 2 0 7 3 2
6 3 4 4 9 7 9 9 8 9 4 6 6 1 7
2 7 3 2 4 7 0 8 9 8 3 0 7 7 7

409 399 386 302 421 279 315 458 385 375 324 258 415 429 374
399 302 353 267 436 240 295 426 342 306 241 210 343 305 258
329 268 297 270 303 167 214 373 305 294 272 215 315 355 258
235 245 224 196 252 194 171 353 283 265 194 211 292 245 263
330 291 251 182 361 253 232 470 292 299 237 266 364 312 312
417 278 270 257 371 195 280 440 305 363 259 188 329 307 262
354 201 215 209 291 113 194 311 197 200 185 126 254 253 199
411 322 387 341 436 190 346 384 392 384 349 211 322 394 263
372 296 299 254 400 223 291 482 413 394 325 274 373 372 316
351 286 217 197 363 162 297 402 328 399 328 263 286 305 257
308 268 335 268 235 176 233 367 247 256 238 112 322 253 229
258 293 286 236 219 160 188 300 250 249 235 173 276 259 235
307 330 276 216 317 235 191 391 311 312 235 253 357 327 299
316 323 264 205 361 234 244 428 307 314 244 230 325 315 292
287 265 191 165 280 140 242 359 222 254 197 205 265 268 249


```
Консоль отладки Microsoft Visual Studio
8 8 0 9 2 3 4 4 6 8 9 2 8 7 6
8 2 4 9 3 2 0 6 8 2 0 7 9 8 6
0 8 3 1 7 8 5 2 5 6 5 0 6 4 4
1 5 4 0 7 3 2 0 1 2 4 1 3 9 9
2 0 6 0 6 0 6 2 5 3 7 4 9 8 9
4 1 0 2 6 4 0 9 8 8 0 7 6 9 5
3 5 2 0 4 1 1 6 8 8 0 1 7 4 1
7 8 0 9 2 9 7 5 8 6 0 4 6 8 1
2 4 9 1 5 7 8 5 5 5 4 2 4 8 9
5 1 4 3 2 4 9 7 4 9 2 9 2 8 2
9 2 2 1 9 9 0 6 1 2 1 1 8 1 7
6 9 0 1 3 7 0 1 0 7 5 4 6 2 3
3 6 3 3 5 2 2 4 1 5 8 5 7 4 8
5 3 1 0 1 2 4 1 7 5 5 9 8 7 8
9 0 0 2 2 0 1 7 9 6 7 9 1 3 1

4 8 8 5 2 2 9 7 2 1 5 0 6 1 7
5 8 7 9 8 1 0 0 9 3 2 5 6 9 5
5 0 3 0 6 1 3 8 6 0 3 8 4 0 2
3 3 6 3 5 3 2 0 5 4 4 3 1 5 0
7 6 5 7 1 0 1 9 0 3 4 6 7 6 5
2 1 9 7 1 2 7 5 8 8 8 0 2 4 0
2 4 0 1 6 0 4 2 4 7 8 4 2 8 3
8 1 1 3 8 2 4 2 6 8 1 0 6 5 2
8 1 4 5 9 0 5 8 4 0 1 1 3 9 3
9 5 0 1 0 0 3 8 1 7 9 2 3 4 5
0 6 3 0 2 7 3 8 4 4 2 9 8 7 9
2 7 1 1 6 2 1 5 5 7 3 9 2 3 1
9 4 9 4 8 7 3 6 0 0 2 0 7 3 2
6 3 4 4 9 7 9 9 8 9 4 6 6 1 7
2 7 3 2 4 7 0 8 9 8 3 0 7 7 7

409 399 386 302 421 279 315 458 385 375 324 258 415 429 374
399 302 353 267 436 240 295 426 342 306 241 210 343 305 258
329 268 297 270 303 167 214 373 305 294 272 215 315 355 258
235 245 224 196 252 194 171 353 283 265 194 211 292 245 263
330 291 251 182 361 253 232 470 292 299 237 266 364 312 312
417 278 270 257 371 195 280 440 305 363 259 188 329 307 262
354 201 215 209 291 113 194 311 197 200 185 126 254 253 199
411 322 387 341 436 190 346 384 392 384 349 211 322 394 263
372 296 299 254 400 223 291 482 413 394 325 274 373 372 316
351 286 217 197 363 162 297 402 328 399 328 263 286 305 257
308 268 335 268 235 176 233 367 247 256 238 112 322 253 229
258 293 286 236 219 160 188 300 250 249 235 173 276 259 235
307 330 276 216 317 235 191 391 311 312 235 253 357 327 299
316 323 264 205 361 234 244 428 307 314 244 230 325 315 292
287 265 191 165 280 140 242 359 222 254 197 205 265 268 249
```

Мал. 4. Результат виконання програми №2

Висновок: навчився працювати з масивами в C.