

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра ЕОМ



**Звіт**  
**до лабораторної роботи № 3**  
**з дисципліни: «Програмування, частина 2 (ООП)»**  
**на тему: «ЗАСОБИ РОБОТИ З ДИНАМІЧНОЮ ПАМ'ЯТТЮ. ДИНАМІЧНІ**  
**МАСИВИ»**  
**Варіант № 14**

Підготував: Мишак М.А.  
студент групи КІ-103

Перевірів:  
Ст. викладач  
Каф. ЕОМ  
Гузинець Н.В

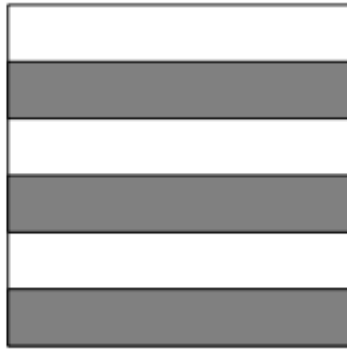
**Мета:** познайомитися із динамічними масивами.

### **Теоретичний матеріал**

В C++ об'єкти можна розміщати статично – під час компіляції, або динамічно – під час виконання програми, шляхом виклику функцій зі стандартної бібліотеки. Основна відмінність у використанні даних методів – в їхній ефективності та гнучкості. Статичне розміщення більш ефективне, так як виділення пам'яті відбувається до виконання програми, проте воно менш гнучке, тому що необхідно наперед знати тип і розмір об'єкту. Задачі, в яких необхідно зберігати та обробляти наперед не відому кількість елементів, зазвичай потребують динамічного виділення пам'яті.

Динамічне виділення пам'яті у мові C++ здійснюється за допомогою оператора `new`. Оператор `new` здійснює пошук неперервної області пам'яті в області пам'яті, що зветься некерована куча. Некерована куча – це структура даних за допомогою якої реалізована пам'ять, що може бути виділена динамічно в ході виконання програми, а також це область пам'яті, зарезервована під цю структуру. З іншої сторони куча - це довгий відрізок адрес пам'яті, поділений на блоки різних розмірів, що йдуть підряд. Пам'ять у кучі поділяється на заняту і вільну. Перед початком роботи програми вся пам'ять у кучі позначається як вільна. При виклику оператора динамічного виділення пам'яті у кучі відбувається пошук неперервного сегменту вільної пам'яті заданого розміру. Час такого пошуку є значним і займає більшу частину часу, що необхідна для виконання операції динамічного виділення пам'яті. Якщо такий сегмент було знайдено в кучі, то він помічається як зайнятий і програмі повертається адреса його початку, інакше – програмі повертається ознака відсутності такого сегменту в пам'яті, найчастіше `NULL`. Якщо в ході виконання програми значення адреси початку цього сегменту втрачається, то заняту пам'ять звільнити буде неможливо. Якщо ця ситуація проявляється неодноразово, то це може призвести до вичерпання вільної пам'яті в системі. Коли динамічно виділена область пам'яті стає непотрібною, то її потрібно звільнити за допомогою оператора звільнення динамічно виділеної пам'яті. При його виклику область пам'яті на яку вказує вказівник і яка була попередньо динамічно виділена з кучі позначається як вільна і її можна буде в подальшому використовувати заново.

**Завдання:** Задано квадратну матрицю, всі елементи якої рівні одиниці. Написати функцію `void func (int** arr, int n)`, котра заповнює заштриховану область матриці (згідно варіанту) нулями. Розмір масиву вводиться з клавіатури.



**Лістинги (тексти) програм:**

```
#include <iostream>
```

```
using namespace std;
```

```
void func(int** arr, int n) {  
    for (int i = 0; i < n; i++) {  
        if (i % 2 == 1) {  
            for (int j = 0; j < n; j++) {  
                arr[i][j] = 0;  
            }  
        }  
    }  
    for (int i = 0; i < n; i++) {  
        for (int j = 0; j < n; j++) {  
            cout << arr[i][j] << " ";  
        }  
        cout << endl;  
    }  
}
```

```
int main() {  
    setlocale(LC_ALL, "uk_UA");  
    int n;  
  
    cout << "Введіть розмір матриці: ";  
    cin >> n;  
    cout << "-----" << endl;
```

```

int** arr = new int* [n];
for (int i = 0; i < n; i++) {
    arr[i] = new int[n];
    for (int j = 0; j < n; j++) {
        arr[i][j] = 1;
    }
}

func(arr, n);

for (int i = 0; i < n; i++) {
    delete[] arr[i];
}
delete[] arr;

return 0;
}

```

## Результати виконання програм:

```

Консоль отладки Microsoft V
Введіть розмір матриці: 7
-----
1 1 1 1 1 1 1
0 0 0 0 0 0 0
1 1 1 1 1 1 1
0 0 0 0 0 0 0
1 1 1 1 1 1 1
0 0 0 0 0 0 0
1 1 1 1 1 1 1

E:\VS22\lab3C++\x64\Debug\lab3C++.exe (процесс 28356) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

```

**Висновок:** на даній лабораторній роботі я ознайомився із динамічними масивами мови програмування C++. На основі цих знань створив програму, яка задає квадратну матрицю, в якій елементи дорівнюють одиниці, а елементи не парного рядка дорівнюють нулю.