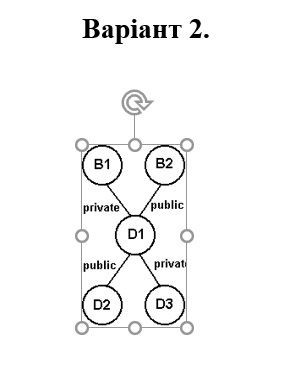
**Лабораторна робота №4**

**Умова:**

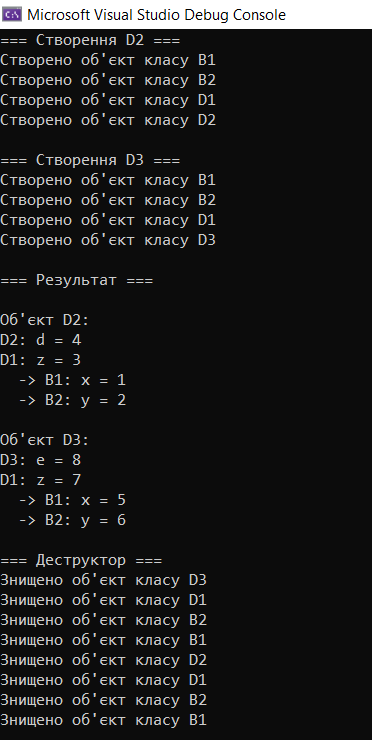
Необхідно побудувати ієрархію класів що відповідає схемі наслідування, наведеній у варіанті завдання. Кожен клас повинен містити конструктор-ініціалізатор, і функцію *show()* для виведення значень.



Код програми:

|  |  |
| --- | --- |
| **Main.cpp**  //Варіант 2  #include "D2.h"  #include "D3.h"  int main() {  setlocale(0, "UKR");  cout << "=== Створення D2 ===" << endl;  D2 obj1(1, 2, 3, 4);  cout << "\n=== Створення D3 ===" << endl;  D3 obj2(5, 6, 7, 8);  cout << "\n=== Результат ===" << endl;  cout << "\nОб'єкт D2:" << endl;  obj1.show();  cout << "\nОб'єкт D3:" << endl;  obj2.show();  cout << "\n=== Деструктор ===" << endl;  return 0;  }  **B1.h**  #pragma once  #include <iostream>  using namespace std;  class B1 {  protected:  int x;  public:  B1(int a) : x(a) {  cout << "Створено об'єкт класу B1" << endl;  }  ~B1() {  cout << "Знищено об'єкт класу B1" << endl;  }  void show() {  cout << "B1: x = " << x << endl;  }  };  **B2.h**  #pragma once  #include <iostream>  using namespace std;  class B2 {  protected:  int y;  public:  B2(int b) : y(b) {  cout << "Створено об'єкт класу B2" << endl;  }  ~B2() {  cout << "Знищено об'єкт класу B2" << endl;  }  void show() {  cout << "B2: y = " << y << endl;  }  }; | #pragma once  #include "B1.h"  #include "B2.h"  class D1 : private B1, public B2 {  protected:  int z;  public:  D1(int a, int b, int c) : B1(a), B2(b), z(c) {  cout << "Створено об'єкт класу D1" << endl;  }  ~D1() {  cout << "Знищено об'єкт класу D1" << endl;  }  void show() {  cout << "D1: z = " << z << endl;  cout << " -> ";  B1::show();  cout << " -> ";  B2::show();  }  };  **D2.h**  #pragma once  #include "D1.h"  class D2 : public D1 {  int d;  public:  D2(int a, int b, int c, int d\_) : D1(a, b, c), d(d\_) {  cout << "Створено об'єкт класу D2" << endl;  }  ~D2() {  cout << "Знищено об'єкт класу D2" << endl;  }  void show() {  cout << "D2: d = " << d << endl;  D1::show();  }  };  **D3.h**  #pragma once  #include "D1.h"  class D3 : private D1 {  int e;  public:  D3(int a, int b, int c, int e\_) : D1(a, b, c), e(e\_) {  cout << "Створено об'єкт класу D3" << endl;  }  ~D3() {  cout << "Знищено об'єкт класу D3" << endl;  }  void show() {  cout << "D3: e = " << e << endl;  D1::show();  }  }; |

**Результат роботи програми:**

****