

## Лабораторна робота №7. Структуровані типи даних

*Тема.* Робота із структурованими типами даних.

*Мета:* придбати практичні навички щодо розроблення програм із застосуванням структур.

*Теми для попереднього опрацювання:*

- структури, доступ до членів структури;
- покажчики на структури;
- масиви структур;
- суміші (об'єднання) та перерахування.

*Загальні відомості*

**Структура** – складений тип даних (створюється програмістом); являє собою набір змінних різних типів, які об'єднані загальною назвою. Оголошення структури призводить до утворення шаблону, який використовується для створення об'єктів структури.

Змінні, що утворюють структуру, називаються членами структури. Члени структури також часто називаються елементами або полями. Як правило, члени структури пов'язані один з одним. Наступний фрагмент програмного коду оголошує шаблон структури, що визначає ім'я та адресу. Ключове слово *struct* повідомляє компілятору про оголошення структури.

```
struct addr
{
    char name[30];
    char street [40];
};
```

Оголошення обов'язково завершується крапкою з комою, оскільки оголошення структури – це оператор. Ім'я структури (у прикладі – *addr*) ідентифікує структуру даних і є специфікатором типу.

Для оголошення змінної, відповідної до даної структури, слід написати:

```
struct addr addr_info
```

Структура не може містити екземпляри самої себе, але можна визначити покажчики на структуру.

Для доступу до елемента структури необхідно вказати її ім'я, поставити крапку і вказати ім'я потрібного елемента. Для доступу до елемента структури через покажчик на об'єкт замість крапки використовується

оператор «стрілка».

**Об'єднання (union)** – це складений тип даних; являє собою набір змінних різних типів, але які перебувають в одній і тій же області пам'яті. Пам'ять виділяється такого об'єму, щоб її було достатньо для зберігання найбільшого члена.

Для оголошення об'єднань використовується ключове слово *union*.

Приклад шаблону суміші з іменем типу *Telements*:

```
union Telements
{
    int number;
    char symbol;
    char *pointer;
};
```

Приклад оголошення даних типу суміші:

```
union Telements dat, mas[5], *pu;
```

Для доступу до членів об'єднання використовуються ті ж синтаксичні конструкції, що і для структури (крапка і стрілка).

Приклад доступу до елементів об'єднання:

```
dat.number=57;
mas[2].symbol='A';
pu=&dat;
x=pu->number;
```

### **Основне завдання:**

Розробити структуру з трьох елементів (полів), згідно з індивідуальним завданням. Створити 4 структурні змінні і заповнити їх значеннями, при цьому:

- чисельні поля генеруються за допомогою функції *rand()*;
- рядкові поля подані вводяться з клавіатури;
- числові поля мають бути згенеровані випадковим чином в прийнятному діапазоні (напр. Рік народження не може бути – «12121»)
- після заповнення структурні змінні вивести на екран.

### **Індивідуальні завдання.**

1. **Завдання з автомобілями:** Створіть структуру **Car**, яка містить поля для року випуску, моделі автомобіля та кольору.
2. **Завдання з працівниками:** Створіть структуру **Employee**, яка містить поля для заробітної плати (число, яке генерується випадковим чином), прізвища працівника (рядок) та його посади (рядок).
3. **Завдання з квартирами:** Створіть структуру **Apartment**, яка містить поля для площі (число, яке генерується випадковим чином), номеру

квартири (рядок) та адреси (рядок).

4. **Завдання з продуктами:** Створіть структуру **Product**, яка містить поля для ціни (число, яке генерується випадковим чином), назви продукту (рядок) та категорії (рядок).
5. **Завдання з книгами:** Створіть структуру **Book**, яка містить поля для року видання (число, яке генерується випадковим чином), назви книги (рядок) та імені автора (рядок).
6. **Завдання з користувачами:** Створіть структуру **User**, яка містить поля для віку (число, яке генерується випадковим чином), імені користувача (рядок) та електронної пошти (рядок).
7. **Завдання з замовленнями:** Створіть структуру **Order**, яка містить поля для кількості одиниць (число, яке генерується випадковим чином), назви товару (рядок) та ідентифікатора замовлення (рядок).
8. **Завдання з клієнтами:** Створіть структуру **Client**, яка містить поля для номера телефону (рядок), прізвища клієнта (рядок) та адреси (рядок).
9. **Завдання з рестораном:** Створіть структуру **Dish**, яка містить поля для ціни за порцію (число, яке генерується випадковим чином), назви страви (рядок) та інгредієнтів (рядок).
10. **Завдання з фільмами:** Створіть структуру **Movie**, яка містить поля для тривалості (число, яке генерується випадковим чином), назви фільму (рядок) та режисера (рядок).
11. **Завдання з комп'ютерами:** Створіть структуру **Computer**, яка містить поля для кількості процесорів (число, яке генерується випадковим чином), моделі комп'ютера (рядок) та обсягу оперативної пам'яті (рядок).
12. **Завдання з меблями:** Створіть структуру **Furniture**, яка містить поля для матеріалу (рядок), типу меблів (рядок) та виробника (рядок).
13. **Завдання з фруктами:** Створіть структуру **Fruit**, яка містить поля для кількості вітамінів (число, яке генерується випадковим чином), назви фрукту (рядок) та кольору шкірки (рядок).
14. **Завдання з музикантами:** Створіть структуру **Musician**, яка містить поля для імені музиканта (рядок), інструменту (рядок) та жанру (рядок).
15. **Завдання з тваринами:** Створіть структуру **Animal**, яка містить поля для кількості ніг (число, яке генерується випадковим чином), назви тварини (рядок) та середовища обитання (рядок).
16. **Завдання з кольорами:** Створіть структуру **Color**, яка містить поля для назви кольору (рядок), коду кольору (рядок) та назви виробника фарби (рядок).

17. **Завдання з напоями:** Створіть структуру **Drink**, яка містить поля для кількості калорій (число, яке генерується випадковим чином), назви напою (рядок) та об'єму (рядок).
18. **Завдання з канцелярськими товариствами:** Створіть структуру **Stationery**, яка містить поля для ціни (число, яке генерується випадковим чином), назви товару (рядок) та виробника (рядок).
19. **Завдання з країнами:** Створіть структуру **Country**, яка містить поля для кількості населення (число, яке генерується випадковим чином), назви країни (рядок) та площі (рядок).
20. **Завдання з спортивними командами:** Створіть структуру **Team**, яка містить поля для кількості перемог (число, яке генерується випадковим чином), назви команди (рядок) та тренера (рядок).

*Додаткові вимоги виконання завдання:*

– звіт має бути виконаний згідно з вимогами до оформлення лабораторних робіт;

*Контрольні питання.*

1. Як виконати доступ до окремих елементів структури?
2. Чи можна вміст однієї структури привласнити іншій того ж типу, використовуючи звичайний оператор присвоювання?
3. Коли необхідно використовувати покажчики на структури?
4. Чи може бути членом структури інша структура?
5. Чим відрізняється об'єднання від структури?
6. Що дозволяє визначити оператор *sizeof*? Чим він корисний?
7. Як працює функція *rand()* ?