

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
«НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
Інститут телекомунікацій, радіоелектроніки та електронної техніки
Кафедра «Радіоелектронні пристрої та системи»



Звіт з лабораторної роботи № 18
«Програмування, частина 2»

Підготував:
ст. гр. ІХ-11
Диркавець Максим
Перевірив:
Асистент каф РЕПС
Чайковський
І.Б.

Тема: Структури та об'єднання даних

Мета роботи: ознайомитися з поняттями структури та об'єднання даних , навчитися їх використовувати у процесі програмування.

Хід роботи

1. Ознайомитися з теоретичними відомостями.
2. Здійснити виконання прикладів, представлених у теоретичних відомостях, після чого представити скріни їх коду та результати виконання у звіті.

Приклад 1

```
#include <stdio.h>

int main(void) {
    struct {
        int a;
        int b;
    } x, y;
    x.a = 10;
    y = x;
    printf("%d", y.a); return 0; }
```

A screenshot of a terminal window with a black background. The number '10' is displayed in white text, indicating the output of the program shown in the previous code block.

Приклад 2

```
#include <stdio.h>

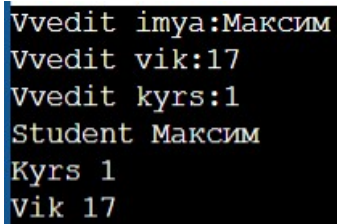
struct student {
    char name[30];
    int kurs;
    int age;
};

int main() {
```

```

    struct student stud1;
    printf("Vvedit imya:");
    fgets(stud1.name, 30, stdin);
    printf("Vvedit vik:");
    scanf("%d", &stud1.age);
    printf("Vvedit kyrs:");
    scanf("%d", &stud1.kurs);
    printf("Student %s", stud1.name);
    printf("Kyr %d\n", stud1.kurs);
    printf("Vik %d\n", stud1.age);
    return 0;
}

```



```

Vvedit imya:Максим
Vvedit vik:17
Vvedit kyrs:1
Student Максим
Kyr 1
Vik 17

```

Приклад 3

```

#include <stdio.h>

struct student {
    char name[30];
    int kurs;
    int age;
};

int main() {
    struct student stud[10];
    int i, n;
    printf("Kilkiet studentiv:");

```

```

scanf("%d", &n);
for(i = 0; i < n; i++) {
    printf("Vvedit imya:");
    scanf("%s", stud[i].name);
    printf("Vvedit vik:");
    scanf("%d", &stud[i].age);
    printf("Vvedit kurs:");
    scanf("%d", &stud[i].kurs);
}
for(i = 0; i < n; i++) {
    printf("Student %s\n", stud[i].name);
    printf("Kurs %d\n", stud[i].kurs);
    printf("Vik %d\n", stud[i].age);
}
return 0;
}

```

```

Student максим
Kurs 1
Vik 17

```

3. Написати програму для виведення нижчепредставленої інформації шляхом використання структури. Ім'я, вага, висота, вік – вводяться з клавіатури (вказати довільні дані). Інформація про працівника Ім'я -----
----- Вага Висота Вік 7

```

#include <stdio.h>

struct Employee {
    char name[50];
    float weight;
    float height;

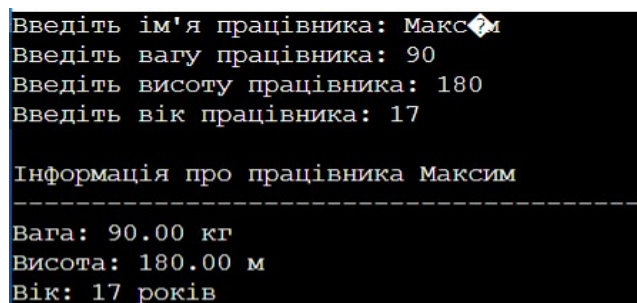
```

```

    int age;
};

int main() {
    struct Employee emp;
    printf("Введіть ім'я працівника: ");
    scanf("%s", emp.name);
    printf("Введіть вагу працівника: ");
    scanf("%f", &emp.weight);
    printf("Введіть висоту працівника: ");
    scanf("%f", &emp.height);
    printf("Введіть вік працівника: ");
    scanf("%d", &emp.age);
    printf("\nІнформація про працівника %s\n", emp.name);
    printf("-----\n");
    printf("Вага: %.2f кг\n", emp.weight);
    printf("Висота: %.2f м\n", emp.height);
    printf("Вік: %d років\n", emp.age);
    return 0;
}

```



```

Введіть ім'я працівника: Макс
Введіть вагу працівника: 90
Введіть висоту працівника: 180
Введіть вік працівника: 17

Інформація про працівника Максим
-----
Вага: 90.00 кг
Висота: 180.00 м
Вік: 17 років

```

4. Оформити звіт.

Висновок: У цій роботі ми ознайомилися з поняттями структур та об'єднань даних у мові програмування. Структури дозволяють об'єднувати кілька типів

даних в один складний тип, що полегшує організацію та обробку даних. Ми практично виконали завдання з введення та виведення інформації про працівника за допомогою структури, що дозволило нам використати одну зручну структуру для зберігання різних атрибутів працівника. Це дозволило зробити програму більш структурованою та зрозумілою.