

Міністерство освіти і науки України
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп’ютерних технологій
Кафедра комп’ютерних наук

Звіт
з лабораторних робіт
з дисципліни «Алгоритми та структури даних»

Виконав студент 1-го курсу, групи ІПЗ-23-1 спеціальності 121 “Інженерія програмного забезпечення”

Сушко О. Ю.

Керівник

Талавер О. В.

Житомир – 2024

					ДУ «Житомирська політехніка».24.121.12.000 – Лр1			
Змн.	Арк.	№ докум.	Підпис	Дата	Звіт з лабораторної роботи ФІКТ Гр. ІПЗ-23-1[2]			
Розроб.	Сушко О.Ю.							
Перевір.	Талавер О.В.							
Керівник								
Н. контр.								
Зав. каф.								
					Літ.	Арк.	Аркушів	
						1	5	

Лабораторна робота № 1

Робота з базовими типами даних

Мета роботи: отримати практичні навички по роботі з базовими типами даних (простими і складними типами даних).

Завдання 1. Записати і заповнити структуру даних зберігання поточного часу (включаючи секунди) і дату в найбільш компактному вигляді. Визначити обсяг пам'яті, яку займає змінна даного типу. Порівняти зі стандартною структурою tm (time.h). Вивести вміст структури в зручному вигляді для користувача на дисплей.

Лістинг програми:

```
#include <stdio.h>
#include <stdint.h>
#include <time.h>

struct CompactDateTime {
    uint8_t seconds;
    uint8_t minutes;
    uint8_t hours;
    uint8_t day;
    uint8_t month;
    uint16_t year;
};

void printCompactDateTime(struct CompactDateTime datetime) {
    printf("%02d:%02d:%02d %02d.%02d.%04d\n", datetime.hours, datetime.minutes, datetime.seconds, datetime.day, datetime.month, datetime.year);
}

int main() {
    printf("Розмір структури CompactDateTime: %zu байт\n", sizeof(struct CompactDateTime));
    printf("Розмір структури tm: %zu байт\n", sizeof(struct tm));
    printf("\n");

    time_t rawtime;
    struct tm timeinfo;
    time(&rawtime);
    localtime_s(&timeinfo, &rawtime);

    struct CompactDateTime currentDateTime = { timeinfo.tm_sec, timeinfo.tm_min,
    timeinfo.tm_hour, timeinfo.tm_mday, timeinfo.tm_mon + 1, timeinfo.tm_year + 1900 };

    printf("Поточний час і дата: ");
    printCompactDateTime(currentDateTime);

    return 0;
}
```

Результат виконання програми:

		Сушко О.Ю.			ДУ «Житомирська політехніка».24.121.12.000 – Лр1	Арк.
		Талавер О.В.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

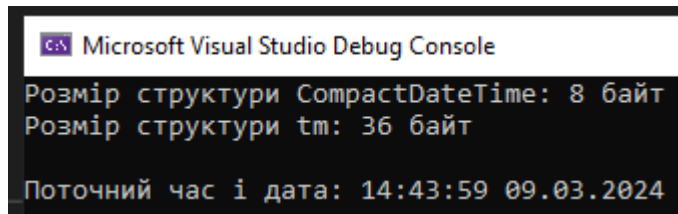


Рис 1. Результат виконання програми завдання 1.

Завдання 2. Реалізувати введення цілочисельного значення типу signed short. Визначити знак і значення, використовуючи: 1) структури даних та об'єднання; 2) побітові логічні операції.

Лістинг програми:

```
#include <stdio.h>

union SignedShort {
    signed short value;
    struct {
        unsigned short sign : 1;
        unsigned short magnitude : 15;
    } parts;
};

int main() {
    union SignedShort num;

    printf("Введіть ціле число типу signed short: ");
    scanf_s("%hd", &num.value);

    printf("Знак: %d\n", num.parts.sign);
    printf("Значення: %d\n", num.parts.magnitude);

    return 0;
}
```

Результат виконання програми:

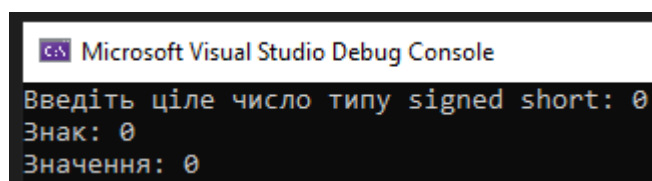


Рис 2. Результат виконання програми завдання 2.

Завдання 3. Виконати операції: а) $5 + 127$; б) $2 - 3$; в) $-120 - 34$; г) (unsigned char) (-5) ; д) $56 \& 38$; е) $56 | 38$. Всі значення (константи) повинні зберігатися в змінних типу signed char. Виконати перевірку результату в ручну. Пояснити результат, використовуючи двійкову систему числення.

Лістинг програми:

```
#include <stdio.h>
```

		Сушко О.Ю.			ДУ «Житомирська політехніка».24.121.12.000 – Лр1	Арк.
		Талавер О.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		3

```

int main() {
    signed char a = 5, b = 127, c = -120, d = -5;
    signed char result;

    result = a + b;
    printf("а) Результат: %d\n", result);

    result = 2 - 3;
    printf("б) Результат: %d\n", result);

    result = c - 34;
    printf("в) Результат: %d\n", result);

    result = (unsigned char)(d);
    printf("г) Результат: %d\n", result);

    result = 56 & 38;
    printf("д) Результат: %d\n", result);

    result = 56 | 38;
    printf("е) Результат: %d\n", result);

    return 0;
}

```

Результат виконання програми:

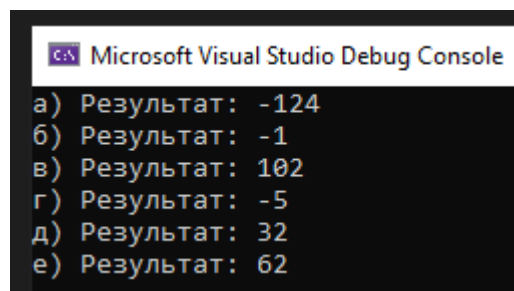


Рис 3. Результат виконання програми завдання 3.

Завдання 4. Записати і заповнити структуру даних (об'єднання) для зберігання дійсного числа типу float в найбільш компактному вигляді. Реалізувати відображення на дисплей: 1) значення побитово; 2) значення побайтово; 3) знака, мантиси і ступінь значення. Виконати перевірку результату в ручну. Визначити обсяг пам'яті, яку займає змінна користувачького типу.

Лістинг програми:

```

#include <stdio.h>
#include <stdint.h>

union FloatRepresentation {
    float value;
    struct {
        uint32_t mantissa : 23;
        uint32_t exponent : 8;
        uint32_t sign : 1;
    } parts;
};

int main() {

```

		Сушко О.Ю.			ДУ «Житомирська політехніка».24.121.12.000 – Лр1	Арк.
		Талавер О.В.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

```

union FloatRepresentation num;

printf("Введіть дійсне число типу float: ");
scanf_s("%f", &num.value);

printf("Побітове представлення: %08x\n", *((unsigned int*)&num.value));
printf("Побайтове представлення: %02x %02x %02x %02x\n", *((unsigned char*)&num.value),
*((unsigned char*)&num.value) + 1), *((unsigned char*)&num.value) + 2), *((unsigned char*)&num.value) + 3));
printf("Знак: %d, Мантиса: %u, Ступінь: %u\n", num.parts.sign, num.parts.mantissa,
num.parts.exponent);

return 0;
}

```

Результат виконання програми:

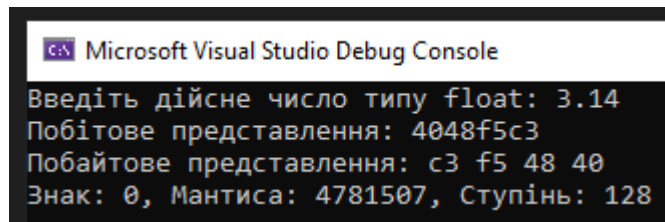


Рис 4. Результат виконання програми завдання 4.

Посилання на GitHub репозиторій: <https://github.com/olexandryipz/Lab1Algo>

Висновки: під час виконання лабораторної роботи я отримав практичні навички по роботі з простими і складними базовими типами даних.

		Сушко О.Ю.			ДУ «Житомирська політехніка».24.121.12.000 – Пр1	Арк.
		Талавер О.В.				5
Змн.	Арк.	№ докум.	Підпис	Дата		