

Міністерство освіти і науки України
Національний університет “Львівська політехніка”



Лабораторна робота №6А
з дисципліни
«Програмування частина 2»

Виконав:
Студент групи АП-11
Братейко Вадим

Прийняв:
Чайковський І.Б.

Львів 2024

Тема роботи: Загальна структура програми мовою C, дослідження використання функцій введення та виведення даних.

Мета роботи: Дослідження структури і використання функцій введення та виведення даних у програмах мовою C.

Приклад 1

```
#include <stdio.h>

int main() {
    int integer = 10;
    float floating = 3.14;
    char character = 'A';
    long long_num = 123456789;

    // Виведення значень з різними модифікаторами специфікації перетворення
    printf("Integer: %d\n", integer);           // integer без модифікатора
    printf("Float: %f\n", floating);           // floating без модифікатора
    printf("Character: %c\n", character);       // character без модифікатора
    printf("Long: %ld\n", long_num);           // long_num з модифікатором l для
long
}
```

```
Integer: 10
Float: 3.140000
Character: A
Long: 123456789
```

Приклад 2

```
#include <stdio.h>

int main() {
    char symbols[10] = {'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'};
    int x;

    printf("Символ\tДесяткове\tВісімкове\tШістнадцяткове\n");
    for (x = 0; x < 10; x++) {
        printf("%c\t%d\t\t%o\t\t%x\n", symbols[x], symbols[x], symbols[x],
symbols[x]);
    }
}
```

Символ	Десяткове	Вісімкове	Шістнадцяткове
A	65	101	41
B	66	102	42
C	67	103	43
D	68	104	44
E	69	105	45
F	70	106	46
G	71	107	47
H	72	110	48
I	73	111	49
J	74	112	4a

Приклад 3

```

#include <stdio.h>
int main() {
    // Оголошення змінних
    float price[5], quantity[5];
    float total[5];
    int x;

    // Введення цін та кількостей товарів з клавіатури
    printf("Введіть ціни та кількості товарів:\n");
    for (x = 0; x < 5; x++) {
        printf("Товар %d:\n", x + 1);
        printf("Ціна: ");
        scanf("%f", &price[x]);
        printf("Кількість: ");
        scanf("%f", &quantity[x]);
        total[x] = price[x] * quantity[x];
    }

    // Виведення результатів у вигляді таблиці
    printf("\n%-10s%-10s%-10s%-10s\n", "Товар", "Ціна", "Кількість", "Сума");
    for (x = 0; x < 5; x++) {
        printf("%-10d%-10.2f%-10.2f%-10.2f\n", x + 1, price[x], quantity[x], total[x]);
    }
}

```

```

Введіть ціни та кількості товарів:
Товар 1:
Ціна: 15
Кількість: 2
Товар 2:
Ціна: 18
Кількість: 5
Товар 3:
Ціна: 12
Кількість: 1
Товар 4:
Ціна: 14
Кількість: 1
Товар 5:
Ціна: 20
Кількість: 3

ТоварЦіна    КількістьСума
1          15.00    2.00    30.00
2          18.00    5.00    90.00
3          12.00    1.00    12.00
4          14.00    1.00    14.00
5          20.00    3.00    60.00

```

Приклад 4

```

#include <stdio.h>
#include <math.h>
#define PI 3.14159

```

```

int main() {
    float radius, circumference, area;

    // Введення радіуса круга з клавіатури
    printf("Введіть радіус круга: ");
    scanf("%f", &radius);

    // Обчислення довжини кола та площі круга
    circumference = 2 * PI * radius;
    area = PI * pow(radius, 2);

    // Виведення результатів
    printf("Довжина кола: %.2f\n", circumference);
    printf("Площа круга: %.2f\n", area);
}

```

```

Введіть радіус круга: 20
Довжина кола: 125.66
Площа круга: 1256.64

```

Приклад 5

```

#include <stdio.h>
#include <math.h>
int main() {
    float a, b, c;
    float discriminant, root1, root2;

    // Введення коефіцієнтів з клавіатури
    printf("Добрий день.\nЗадайте коефіцієнти a, b, і c: ");
    scanf("%f %f %f", &a, &b, &c);

    // Обчислення дискримінанта
    discriminant = b * b - 4 * a * c;

    // Перевірка дискримінанта для визначення коренів
    if (discriminant > 0) {
        // Два різних дійсних корені
        root1 = (-b + sqrt(discriminant)) / (2 * a);
        root2 = (-b - sqrt(discriminant)) / (2 * a);
        printf("За розрахунками в мене вийшли такі корені рівняння: %.2f та %.2f\n",
            root1, root2);
    } else if (discriminant == 0) {
        // Один дійсний корінь
        root1 = -b / (2 * a);
        printf("За моїми розрахунками рівняння має один подвійний корінь: %.2f\n",
            root1);
    } else {
        // Корені у вигляді комплексних чисел
        float realPart = -b / (2 * a);

```

```

float imaginaryPart = sqrt(-discriminant) / (2 * a);
printf("За розрахунками в мене вийшли такі корені рівняння: %.2f + %.2fi та  

%.2f - %.2fi\n", realPart, imaginaryPart, realPart, imaginaryPart);
}
}

```

Добрий день.

Задайте коефіцієнти a, b, i c: 9 12 15

За розрахунками в мене вийшли такі корені рівняння: -0.67 + 1.11i та -0.67 - 1.11i

Приклад 6

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

```
int main() {
```

```
    char ch;
```

```
    printf("Введіть символи. Для завершення натисніть звукову клавішу  

(наприклад, Ctrl+G):\n");
```

```
    while ((ch = getchar()) != 7) { // 7 - ASCII код звукової клавіші (Ctrl+G)
        putchar(ch);
    }
}

```

```

Введіть символи. Для завершення натисніть звукову клавішу (наприклад, Ctrl+G):
22
22
agd^

```

Приклад 7

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

```
#include <math.h>
```

```
int main() {
```

```
    // Задані координати вершин трикутника
```

```
    int k = 5;
```

```
    int x1 = 1, y1 = 1;
```

```
    int x2 = 2 * k, y2 = 2 * k - 1;
```

```
    int x3 = -2 * k, y3 = k + 2;
```

```
    // Обчислення відстані між точками
```

```
    double a = sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2));
```

```
    double b = sqrt(pow(x3 - x2, 2) + pow(y3 - y2, 2));
```

```
    double c = sqrt(pow(x1 - x3, 2) + pow(y1 - y3, 2));
```

```
    // Обчислення півпериметра
```

```
    double p = (a + b + c) / 2;
```

```
    // Обчислення радіуса вписаного кола
```

```
    double s = sqrt(p * (p - a) * (p - b) * (p - c));
```

```
    double r = s / p;
```

```
    // Виведення результатів
```

```
    printf("Периметр трикутника: %.2f\n", a + b + c);
```

```
    printf("Площа трикутника: %.2f\n", s);
```

```
    printf("Радіус вписаного кола: %.2f\n", r);
}

```

```
}  
Периметр трикутника: 44.67  
Площа трикутника: 71.00  
Радіус вписаного кола: 3.18
```

Приклад 8

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

```
int main() {  
    double a = 1000, b = 0.0001;  
    double result;  
    result = (pow(a + b, 3) - (pow(a, 3) + 3 * a * a * b)) / (3 * a * b * b + pow(b, 3));  
    printf("Результат виразу при a = %.4f та b = %.4f: %.4f\n", a, b, result);  
    return 0;  
}
```

```
Результат виразу при a = 1000.0000 та b = 0.0001: 0.9974
```

Контрольні запитання

1) Структура програми на мові C:

Програма на мові C зазвичай складається з принаймні однієї функції, яка має назву `main()`. Це вихідний пункт виконання програми. Крім того, програма може містити інші користувацькі функції, які викликаються з `main()` або з інших функцій.

2) Ідеологія організації операцій введення-виведення в мові C:

У мові C операції введення-виведення організовані через стандартні бібліотечні функції, які включені в заголовний файл `stdio.h`. Ці функції, такі як `printf()` і `scanf()`, дозволяють виконувати виведення і введення даних відповідно.

3) Синтаксис функцій `printf()` і `scanf()`:

```
printf("формат_строки", список_аргументів);  
scanf("формат_строки", &змінні);
```

4) Основні типи форматів при звертанні до функцій `printf()` і `scanf()`:

`%d` - для цілих чисел

`%f` - для дійсних чисел

`%c` - для символів

`%s` - для рядків

5) Модифікатори форматів при звертанні до функцій `printf()` і `scanf()`:

`%` - вказує на початок специфікатора формату

`*` - вказує на пропуск полів в аргументі

6) Відмінності при застосуванні функцій `printf()` і `scanf()`:

`printf()` використовується для виведення даних на екран у вказаному форматі.

`scanf()` використовується для отримання даних з клавіатури у вказаному

форматі.

7)Застосування функцій `getchar()` і `putchar()`:

`getchar()` - отримує наступний символ зі стандартного вводу.

`putchar()` - виводить один символ на стандартний вивід.

Висновок: на цій лабораторній роботі я ознайомився і дослідив структури і використав функцій уведення та виведення даних у програмах мовою C.