

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Інститут телекомунікацій, радіоелектроніки та електронної техніки
кафедра «Радіоелектронні пристрої та системи»



Звіт з лабораторної роботи №ба

з дисципліни «Програмування»

Підготував:
ст. групи АП-11
Фостик Віталій Іванович

Прийняла:
Гордійчук-Бублівська О. В.

Тема:

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

Мета:

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

Теоретичні відомості:

Програма на мові C складається з однієї або більше функцій і хоча б одна з них повинна називатися `main()`. Опис функції складається з заголовку та тіла. Заголовок у свою чергу містить директиви препроцесора типу `#include` і.т.д, що під'єднують бібліотечні файли та специфікують перетворення тексту програми перед компіляцією; а також ім'я функції. Ознакою імені функції служать круглі дужки. Тіло функції поміщається в фігурні дужки та є набором операторів (команд), кожен із яких закінчується символом `“ ; “` - крапка з комою. Елементом програми є коментар - частина тексту програми для пояснення окремих операторів, що входять до її складу. Коментар не впливає на виконання операторів і записується таким чином: `// текст коментарю` або так: `/* текст коментарю */`. В першому випадку коментар має бути єдиним у рядку або в кінці рядка. Другий спосіб дозволяє записувати коментар будь-де в тексті програми не розриваючи лексем.

Завдання:

1. Виконати усі приклади, що наведені в теоретичних відомостях.
2. У звіті зазначити формати, що використовуються функціями `printf()` і `scanf()`.
3. Створити програму, в якій задати числа, що оголошені як типи `int`, `float`, `char`, `long` та вивчити вплив модифікаторів специфікації перетворення для функції `printf()`.
4. Надрукувати в рядок 10 будь-яких символів таблиці ASCII та відповідні їм коди в десятковій, вісімковій, шістнадцятковій системах.
5. Створити програму для розв'язання задачі купівлі товарів за формулою - вартість купівлі дорівнює: ціна товару помножена кількість. Знайти суму купівлі при номенклатурі товарів не менше 5. Вхідні дані задавати: а). під час оголошення змінних, б). введенням із клавіатури використовуючи функцію `scanf()`. Результати оформити у вигляді таблиці.
6. Створити програму обчислення довжини кола та площі круга за радіусом, який задавати введенням із клавіатури.
7. Створити програму обчислення коренів квадратного рівняння. Задачу виконати у вигляді діалогу з введенням набору коефіцієнтів за допомогою клавіатури.
8. Модифікувати програму виведення на друк групи символів із

застосуванням функцій `getchar()` і `putchar()` так, щоб символом припинення роботи програми при введенні був звуковий сигнал - функція `sound(частота)`.

9. Обчислити периметр трикутника, його площу та радіус вписаного кола за заданими координатами його вершин A(1; 1), B(2k; 2k-1), C(-2k; k+2), де k – номер варіанта.

10.

7	$\frac{(a-b)^3 - (a^3)}{b^3 - 3ab^2 - 3a^2b}, \text{ при } a=1000, b=0.0001$
---	--

Виконання роботи:

```
1 #include <stdio.h>
2 void main()
3 {
4     int z;
5     int w;
6     int x=1;
7     int y=2;
8     z=y+x;
9     w=y-x;
10    printf("z=%d w=%d", z,w);
11 }
```

/tmp/90IpRlTAqg.o
z=3 w=1
=== Code Exited With Errors ===

```
1 #define PI 3.14159
2 #include <stdio.h>
3 void main()
4 {
5     int a =5;
6     float b=23.5;
7     int c=31000;
8     printf("%d метрів тканини коштувало %f гривень. \n", a,b);
9     printf("Значення числа пі рівне %f.\n", PI);
10    printf("IBM сумісні комп'ютери набули широкого розповсюдження.\n");
11    printf("%c\n", '$', c);
12 }
```

/tmp/BsOwpUH1p0.o
5 метрів тканини коштувало 23.500000 гривень.
Значення числа пі рівне 3.141590.
IBM сумісні комп'ютери набули широкого розповсюдження.
\$31000
=== Code Exited With Errors ===

```
1 #include <stdio.h>
2 void main()
3 {
4     printf("/d/\n",557);
5     printf("/%2d/\n",557);
6     printf("/%10d/\n",557);
7     printf("/%-10d/\n",557);
8 }
```

/tmp/PhhVuvVsXM.o
/557/
/557/
/ 557/
/557 /
=== Code Exited With Errors ===

```
1 #include <stdio.h>
2 void main() {
3     printf("/f/\n", 6543.21);
4     printf("/e/\n", 6543.21);
5     printf("/%4.2f/\n", 6543.21);
6     printf("/%3.1f/\n", 6543.21);
7     printf("/%10.3f/\n", 6543.21);
8     printf("/%10.3e/\n", 6543.21);
9 }
```

/tmp/m39v737vQD.o
/6543.210000/
/6.543210e+03/
/6543.21/
/6543.2/
/ 6543.210/
/ 6.543e+03/
=== Code Exited With Errors ===

<pre> 1 #include <stdio.h> 2 #define riadok "Чудова мороза" 3 void main(){ 4 printf("/%25/\n", riadok); 5 printf("/%15.s/\n", riadok); 6 printf("/%15.5s/\n", riadok); 7 printf("/%15.5s/\n", riadok); 8 } </pre>	<pre> /tmp/bjMucSQctn.o /%25/ / / Чуд/ /Чуд / === Code Exited With Errors === </pre>
---	---

<pre> 1 #include <stdio.h> 2 void main() { 3 printf("%d\n", 557); 4 printf("%o\n", 557); 5 printf("%x\n", 557); 6 printf("%d\n", -557); 7 printf("%u\n", -557); 8 } </pre>	<pre> /tmp/DkMHUCPZci.o 557 1055 22d -557 4294966739 === Code Exited With Errors === </pre>
--	--

<pre> 1 #include <stdio.h> 2 void main() { 3 int vik; 4 char name [30]; 5 printf("Vash vik?\n"); 6 scanf("%d", &vik); 7 printf("Vvedit vashe imya\n"); 8 scanf("%s", name); 9 printf("Pryvit %s jakomy(iy) %d roki", name, vik); 10 } </pre>	<pre> /tmp/CMhYAmueBp.o Vash vik? 18 Vvedit vashe imya Test Pryvit Test jakomy(iy) 18 roki === Code Exited With Errors === </pre>
--	--

<pre> 1 #include <stdio.h> 2 #define STOP '*' 3 4 void main() { 5 char ch; 6 ch = getchar(); 7 m1: 8 if (ch != STOP) { 9 putchar(ch); 10 ch = getchar(); 11 goto m1; 12 } 13 } 14 </pre>	<pre> /tmp/McGseUZqLt.o 123*ghjghj 123 === Code Exited With Errors === </pre>
---	--

3.

<pre> 1 #include <stdio.h> 2 void main() { 3 int i = 99; 4 float f = 96.14; 5 char c = 'X'; 6 long b = 649851648; 7 printf("Integer: %d\n", i); 8 printf("Float: %f\n", f); 9 printf("Character: %c\n", c); 10 printf("Lond: %ld\n", b); 11 } </pre>	<pre> /tmp/PIrokjsY7l.o Integer: 99 Float: 96.139999 Character: X Lond: 649851648 === Code Exited With Errors === </pre>
--	---

4.

1	#include <stdio.h>	/tmp/pwjrpwrnMZM.o
2		ASCII символи та їх коди:
3	void main() {	Символ Десятковий Вісімковий Шістнадцятковий
4	int i;	
5	char c;	
6		
7	printf("ASCII символи та їх коди: \n");	O 79 117 4f
8	printf("Символ Десятковий Вісімковий Шістнадцятковий\n");	P 80 120 50
9		Q 81 121 51
10	for (i = 0, c = '0'; i < 10; ++i, ++c) {	R 82 122 52
11	printf("\n%c %d %o %x\n", c, c, c, c);	S 83 123 53
12	}	T 84 124 54
13		U 85 125 55
14	}	V 86 126 56
15		W 87 127 57
		X 88 130 58
		=== Code Exited With Errors ===

5.

1	#include <stdio.h>	/tmp/mju8oC6wb0.o
2		Товар Ціна Кількість Сума
3	void main() {	Товар 1 10.50 3 31.50
4	int i, n;	Товар 2 20.00 2 40.00
5	float price, quantity, total = 0;	Товар 3 15.75 5 78.75
6	float prices[] = {10.5, 20, 15.75, 8.99, 12.49};	Товар 4 8.99 1 8.99
7	int quantities[] = {3, 2, 5, 1, 4};	Товар 5 12.49 4 49.96
8	printf("Товар\tЦіна\tКількість\tСума\n");	Загальна сума покупки: 209.20
9	for (i = 0; i < 5; ++i) {	
10	price = prices[i];	Введіть кількість товарів (не менше 5): 5
11	quantity = quantities[i];	Введіть ціну та кількість для кожного товару:
12	total += price * quantity;	Товар Ціна Кількість Сума
13	printf("Товар %d\t%.2f\t%d\t%.2f\n", i + 1, price, quantities[i], price * quantity);	Товар 1: 40 5
14	}	Товар 1 40.00 5 200.00
15	printf("Загальна сума покупки: %.2f\n\n", total);	Товар 2: 10 1
16	total = 0;	Товар 2 10.00 1 10.00
17	printf("Введіть кількість товарів (не менше 5): ");	Товар 3: 15 8
18	scanf("%d", &n);	Товар 3 15.00 8 120.00
19	float dynamic_prices[n];	Товар 4: 15514 7
20	int dynamic_quantities[n];	Товар 4 15514.00 7 108598.00
21	printf("Введіть ціну та кількість для кожного товару: \n");	Товар 5: 10963 85
22	printf("Товар\tЦіна\tКількість\tСума\n");	Товар 5 10963.00 85 931855.00
23	for (i = 0; i < n; ++i) {	Загальна сума покупки: 1040783.00
24	printf("Товар %d: ", i + 1);	
25	scanf("%f %d", &price, &dynamic_quantities[i]);	
26	dynamic_prices[i] = price;	
27	total += price * dynamic_quantities[i];	
28	printf("Товар %d\t%.2f\t%d\t%.2f\n", i + 1, price, dynamic_quantities[i], price * dynamic_quantities[i]);	
29	}	
30	printf("Загальна сума покупки: %.2f\n", total);	
31		
32	}	
33		=== Code Exited With Errors ===

6.

1	#include <stdio.h>	/tmp/PZxKrsK28T.o
2	#include <math.h>	Введіть радіус круга: 5
3		Довжина кола: 31.42
4	#define PI 3.14159	Площа круга: 78.54
5		
6	void main() {	
7	float radius;	
8	printf("Введіть радіус круга: ");	
9	scanf("%f", &radius);	
10		
11	float circle_length = 2 * PI * radius;	
12	float area = PI * pow(radius, 2);	
13		
14	printf("Довжина кола: %.2f\n", circle_length);	
15	printf("Площа круга: %.2f\n", area);	
16		
17	}	
18		=== Code Exited With Errors ===

7.

```
1 #include <stdio.h>
2 #include <math.h>
3
4- void main() {
5     float a, b, c;
6     float discriminant, root1, root2;
7
8     printf("Введіть a, b, c квадратного рівняння (ax^2 + bx + c): \n");
9     printf("a: ");
10    scanf("%f", &a);
11    printf("b: ");
12    scanf("%f", &b);
13    printf("c: ");
14    scanf("%f", &c);
15
16    discriminant = b * b - 4 * a * c;
17
18-    if (discriminant > 0) {
19        root1 = (-b + sqrt(discriminant)) / (2 * a);
20        root2 = (-b - sqrt(discriminant)) / (2 * a);
21
22        printf("Корені квадратного рівняння: %.2f та %.2f\n", root1, root2);
23-    } else if (discriminant == 0) {
24        root1 = root2 = -b / (2 * a);
25
26        printf("Корені квадратного рівняння: %.2f та %.2f (два однакові корені)\n", root1, root2);
27-    } else {
28        float realPart = -b / (2 * a);
29        float imaginaryPart = sqrt(-discriminant) / (2 * a);
30
31        printf("Корені квадратного рівняння: %.2f + %.2fi та %.2f - %.2fi\n", realPart, imaginaryPart,
32              realPart, imaginaryPart);
33    }
34 }
```

/tmp/SHU05ZDAQ.o
Введіть a, b, c квадратного рівняння (ax^2 + bx + c):
a: 9
b: 1
c: 2
Корені квадратного рівняння: -0.06 + 0.471 та -0.06 - 0.471

=== Code Exited With Errors ===

8.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4- void sound() {
5     printf("\a");
6 }
7
8- int main() {
9     char symbol;
10
11    printf("Введіть символ. Натисніть звуковий сигнал (наприклад, клавіша Enter) для завершення: \n");
12
13-    while ((symbol = getchar()) != EOF && symbol != '\n') {
14        putchar(symbol);
15    }
16
17    sound();
18
19    printf("Програма завершилася. \n");
20
21    return 0;
22 }
```

/tmp/JPV70HqV24.o
Введіть символ. Натисніть звуковий сигнал (наприклад, клавіша Enter) для завершення:
gfgfa
gfgfa Програма завершилася.

=== Code Execution Successful ===

9.

```
1 #include <stdio.h>
2 #include <math.h>
3
4 #define k 7
5
6- float distance(float x1, float y1, float x2, float y2) {
7     return sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2));
8 }
9
10- void main() {
11     float xA = 1, yA = 1;
12     float xB = 2 * k, yB = 2 * k - 1;
13     float xC = -2 * k, yC = k + 2;
14     float AB = distance(xA, yA, xB, yB);
15     float BC = distance(xB, yB, xC, yC);
16     float AC = distance(xA, yA, xC, yC);
17     float perimeter = AB + BC + AC;
18     float p = perimeter / 2;
19     float area = sqrt(p * (p - AB) * (p - BC) * (p - AC));
20     float inradius = area / p;
21
22     printf("Периметр трикутника: %.2f\n", perimeter);
23     printf("Площа трикутника: %.2f\n", area);
24     printf("Радіус вписаного кола: %.2f\n", inradius);
25 }
```

/tmp/c3LlwHSLCn.o
Периметр трикутника: 62.98
Площа трикутника: 142.00
Радіус вписаного кола: 4.51

=== Code Exited With Errors ===

10.

```
1 #include <stdio.h>
2 #include <math.h>
3
4- void main() {
5     float a = 1000, b = 0.0001, x;
6     x = (pow(a-b,3) - (pow(a,3))) / (pow(b,3) - 3*pow(a,2)*b);
7     printf("Результат: %f", x);
8 }
9 }
```

/tmp/ZNKZnDCOWM.o
Результат: 1.220703

=== Code Exited With Errors ===

Контрольні запитання:

1. Структура програми на мові C:

- Включення необхідних бібліотек (`#include` директива).
- Об'явлення функцій та змінних.
- Основна функція `main()`, яка виконує основну логіку програми.
- Інші користувацькі функції.
- Повернення значення з функції `main()` за допомогою `return`.

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

- Використання функцій `printf()` та `scanf()` для виведення та введення даних відповідно.
- Взаємодія з консоллю (стандартними потоками введення-виведення).

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

- `printf()`:
```  
printf("формат", список\_аргументів);  
```
- `scanf()`:
```  
scanf("формат", &змінні);  
```

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

- `%d` - десяткове число (int).
- `%f` - десяткове число з плаваючою комою (float).
- `%c` - символ.
- `%s` - рядок.
- `%x` - шістнадцяткове число.
- `%u` - беззнакове десяткове число.
- `%p` - адреса пам'яті.

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

- `%-` - вирівнювання відносно лівого краю.
- `%+` - виведення зі знаком.
- `%0` - заповнення пробілами нульового значення.

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

- `printf()` використовується для виведення даних, а `scanf()` - для введення.
- `printf()` не вимагає амперсанду (&) перед змінною, тоді як `scanf()` вимагає його для передачі адреси змінної.

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

- `getchar()` чекає на введення одного символу з консолі.
- `putchar()` виводить один символ на консоль.

