

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



Звіт з лабораторної роботи №6

з дисципліни «Програмування»

Підготував:  
ст. групи АП-11  
Білоніжко В. А.

Прийняла:  
Чайковський І. Б.

**Тема:**

Загальна структура програми на мові C

**Мета:**

ознайомитися із загальною структурою побудови програм на мові C, навчитися використовувати функції введення та виведення даних.

**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` тощо, 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*/`

**Хід роботи:**

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

**Виконання роботи:**

<pre>1  #include &lt;stdio.h&gt; 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 }</pre>	<pre>z=3 , w=1</pre>
--	----------------------

<pre> 1  #define PI 3.14159 2  #include &lt;stdio.h&gt; 3  void main() 4  { 5      int a=5; 6      float b=23.5; 7      int c=31000; 8      printf("%d метрів тканини вартувало           %f гривень.\n", a,b); 9      printf("Значення числа pi рівне %f           .\n", PI); 10     printf("IBM сумісні коп'ютери набули           широкого розповсюдження.\n"); 11     printf("%c%d\n", '\$', c); 12 }</pre>	<pre> /tmp/gnYokTid20.o 5 метрів тканини вартувало 23.500000 гривень. Значення числа pi рівне 3.141590. IBM сумісні коп'ютери набули широкого розповсюдження. \$31000</pre>
---	---

<pre> 1  #include &lt;stdio.h&gt; 2  void main() 3  { 4      printf("/%d/\n", 557); 5      printf("/%10d/\n", 557); 6      printf("/%-10d/\n", 557); 7  }</pre>	<pre> /tmp/JcwAUwiddo.o /557/ /          557/ /557          /</pre>
---	---

<pre> 1  #include &lt;stdio.h&gt; 2  void main() 3  { 4      printf("/%8d/\n", 21949); 5      printf("/%8d/\n", -1611); 6      printf("/%8d/\n", -8380); 7      printf("/%8d/\n", 2143); 8  }</pre>	<pre> /tmp/ZwUYp4WuPD.o / 21949/ / -1611/ / -8380/ / 2143/</pre>
---	--

1	#include <stdio.h>	/tmp/b2zdiEzCLW.o
2	void main()	/6543.210000/
3	{	/6.543210e+03/
4	printf("/%f/\n", 6543.21);	/6543.21/
5	printf("/%e/\n", 6543.21);	/6543.2/
6	printf("/%4.2f/\n", 6543.21);	/ 6543.210/
7	printf("/%3.1f/\n", 6543.21);	/ 6.543e+03/
8	printf("/%10.3f/\n", 6543.21);	
9	printf("/%10.3e/\n", 6543.21);	
10	}	

1	#include <stdio.h>	/Чудова погода/
2	#define riadok "Чудова погода"	
3	void main()	
4	{	
5	printf("/2s/\n", riadok);	
6	printf("/%15.s/\n", riadok);	
7	}	

1	#include <stdio.h>	/tmp/2PQJBoEMuR.o
2	void main()	557/
3	{	1055/
4	printf("%d/\n", 557);	22d/
5	printf("%o/\n", 557);	-557/
6	printf("%x/\n", 557);	
7	printf("%d/\n", -557);	
8	}	

<pre>1 #include &lt;stdio.h&gt; 2 void main() 3 { 4     int vik; 5     char name[30]; 6     printf("Vash vik?\n"); 7     scanf("%d", &amp;vik); 8     printf("Vvedit vashe imya\n"); 9     scanf("%s", name); 10    printf("Pryvit %s jakomy(iy) %d         rokiv", name, vik); 11 }</pre>	<pre>/tmp/xLKPWfaIHF.o Vash vik? 17 Vvedit vashe imya Bitalik Pryvit Bitalik jakomy(iy) 17 rokiv</pre>
--	--

<pre>1 #include &lt;stdio.h&gt; 2 #include &lt;conio.h&gt; 3 #define STOP '*' 4 void main() 5 { 6     char ch; 7     ch=getchar(); 8     m1: if(ch!=STOP){ 9         putchar(ch); 10        ch=getchar(); goto m1; 11    } 12 }</pre>	<pre>h h t t *</pre>
---	----------------------

### Контрольні запитання:

#### 1. Структура програми на мові С:

Програма на мові С зазвичай складається з функцій, які можуть бути вбудованими або створеними користувачем. У загальному випадку, програма на С має заголовок, включає директиви препроцесора, область оголошень, функції, та може містити також директиви для створення інших типів об'єктів, таких як макроси, структури, та інші.

#### 2. Ідеологія організації операцій введення-виведення в мові С:

Організація операцій введення-виведення в мові С базується на використанні стандартних бібліотек, зокрема `stdio.h`. Для введення використовується функція `scanf()`, а для виведення - `printf()`. Ці функції мають багато можливостей для форматування введених і виведених даних.

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

Синтаксис функції ``printf()``:

```
```c
printf("format_string", arg1, arg2, ...);
```
```

Синтаксис функції ``scanf()``:

```
```c
scanf("format_string", &var1, &var2, ...);
```
```

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

Основні типи форматів включають ``%d`` для цілих чисел, ``%f`` для дійсних чисел, ``%c`` для символів, ``%s`` для рядків тощо.

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

Модифікатори форматів включають ширину поля, точність (у разі змінних типу ``float`` для ``printf()``), та інші.

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

Основна відмінність полягає у тому, що ``printf()`` використовується для виведення даних на екран, тоді як ``scanf()`` - для отримання даних з клавіатури.

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

``getchar()`` використовується для отримання одного символу зі стандартного потоку вводу (зазвичай клавіатури), тоді як ``putchar()`` використовується для виведення одного символу на стандартний потік виводу (зазвичай екран).

### 8. Пояснення змісту і обґрунтування результатів виконаних прикладів:

функції ``printf()`` і ``scanf()`` дозволяють взаємодіяти з користувачем шляхом виведення та отримання даних від нього, тоді як ``getchar()`` і ``putchar()`` дають змогу працювати з одним символом за раз. Всі ці функції важливі для забезпечення коректної роботи програми та взаємодії з користувачем.