

ЛАБОРАТОРНА РОБОТА №5

Символьне введення / виведення даних

Допоміжні відомості, що можуть знадобитись під час виконання практичних завдань.

I/O – Input/Output – Введення/виведення даних

Типи даних

Digits - {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}.

Hexadecimal digits - {0 1 2 3 4 5 6 7 8 9 A B C D E F a b c d e f}.

Lowercase letters - {a b c d e f g h i j k l m n o p q r s t u v w x y z}.

Uppercase letters - {A B C D E F G H I J K L M N O P Q R S T U V W X Y Z}.

Letters - lowercase + uppercase letters.

Alphanumeric characters - Digits + Lowercase letters + Uppercase letters.

Punctuation characters - ! " # \$ % & ' () * + , - . / : ; < = > ? @ [\] ^ _ ` { | } ~

Graphical characters - Alphanumeric characters + Punctuation characters.

Space characters - tab, newline, vertical tab, form feed, carriage return, and space.

Printable characters - Alphanumeric characters, Punctuation characters and Space characters.

Control characters - In ASCII, these characters have octal codes 000 through 037, and 177 (DEL).

Blank characters - spaces + tabs.

Alphabetic characters - Lowercase letters + Uppercase letters.

Функції бібліотеки <ctype.h>

```
1 #include <ctype.h>
2 int isalnum(int c); //Если аргумент функции является либо буквой, либо цифрой, она возвращает ненулевое значение.
3 int isalpha(int c); //Возвращает ненулевое значение, если её аргумент является буквой, в противном случае возвращается нуль.
4 int isblank(int c); //Возвращает true, если c - пробел или горизонтальная табуляция (C99).
5 int iscntrl(int c); //Возвращает true, если c - управляющий символ, такой как <Ctrl+B>.
6 int isdigit(int c); //Возвращает ненулевое значение, если её аргумент является десятичной цифрой, в противном случае возвращается нуль.
7 int isgraph(int c); //Возвращает true, если c - печатаемый символ, отличный от пробела.
8 int islower(int c); //Возвращает true, если c - символ нижнего регистра.
9 int isprint(int c); //Возвращает true, если c - печатаемый символ.
10 int ispunct(int c); //Возвращает true, если c - знак препинания (любой печатаемый символ, отличный от пробела или алфавитно-цифрового символа).
11 int isspace(int c); //Возвращает true, если c - пробельный символ: пробел, новая строка, перевод строки, возврат каретки, вертикальная табуляция, горизонтальная табуляция или, возможно, другой символ, определяемый реализацией
12 int isupper(int c); //Возвращает true, если c - символ верхнего регистра.
13 int isxdigit(int c); //Возвращает true, если c - шестнадцатеричная цифра.
```

Функції введення / виведення даних на екран:

printf() – комплексна функція для виведення даних будь-якого типу;

putchar() – виводить дані типу char, по одному символу (символьний літерал);

puts() – виводить символьні рядки на екран (рядковий літерал).

scanf() – функція, що зчитує дані будь-якого типу;

getchar() – зчитує з клавіатури один символ;

gets() – зчитує з клавіатури рядок символів.

Завдання 1

Напишіть коротку програму, яка буде виводити на екран користувачеві повідомлення з запитом «Ввести будь-які декілька символів». Програма має зчитати ці символи і повторно вивести їх на екран наступним чином:

1) в один рядок з додаванням пробілів між ними, після останнього символу пробіл додаватись не повинен. Наприклад, введено abc => вивід a b c.

Якщо користувач вводить слова з пробілами та/або розділовими знаками, то і в кінці останнього запису, і на місцях пробілів введених користувачем, додаткові пробіли з'являться не повинні. Наприклад (рисунок 1), Do, re, Mi, A! => D o , r e , M i , A !

abc => a b c
Do, re, Mi, A! => D o , r e , M i , A !

Рисунок 1 – Приклад вводу і виводу першого варіанту

2) в один рядок виводиться слово і між його літерами ставляться пробіли, при цьому кожне наступне слово виводиться з нового рядка. Наприклад (рисунок 2), I don't want to study => I

d o n ' t
w a n t
t o
s t u d y

I don't want to study
I don't want to study

Рисунок 2 – Приклад вводу і виводу другого варіанту

У якому саме вигляді виводити – 1 чи 2 тип – має обирати користувач.

Програма не повинна завершуватись після одного разу, а запитувати користувача чи бажає він продовжити, чи вийти.

Завдання 2

Напишіть програму, яка буде виводити букви англійського алфавіту від першої A до останньої Z. Для цього доцільно скористатись циклом for. Алфавіт потрібно вивести наступними способами:

- 1) в рядок з пробілами між літери усі літери верхнього регістру;
- 2) в рядок з пробілами, усі літери нижнього регістру;
- 3) в рядок, літера верхнього регістру + літера нижнього регістру + пробіл, наступні літери і так до кінця алфавіту (а саме Aa b Cc Dd Zz).

Пробілу після останньої літери не має бути.

У якому саме вигляді виводити має обирати користувач.

Завдання 3

Напишіть програму, яка буде всі введені користувачем літери, повторно виводити на екран, але у верхньому регістрі.

Повторіть те ж саме, але навпаки: нехай, програма запрошує у користувача введення літер верхнього регістру, а потім виводить їх маленькими (у нижньому регістрі).

Для користувача потрібно організувати зрозуміле меню та не завершувати програму після однієї дії.

Завдання 4

Напишіть програму, яка буде перевіряти чи є введений користувачем символ буквою алфавіту чи ні, та виводитиме відповідні повідомлення (наприклад: Enter symbol %c is an alphabet).

У випадку коли користувач вводить не літеру з алфавіту, а символ, то програма має виконувати перевірку за функцією `isspace()` і також виводити відповідне повідомлення.

Якщо введений користувачем символ не підходить в перший чи другий варіанти програма має спробувати перевірити його втретє на щось за вашим бажанням (оберіть зі списку функцій в лекції).

Якщо символ не підходить під жодну перевірку має виводитись відповідне повідомлення.

В кінці має виконуватись запит чи продовжити / чи завершити роботу.