

Контрольні запитання до лабораторної роботи № 1:

1. Яка структура простої програми на мові С?

Структура простої програми, яка щось виконує, буде складатися з директив препроцесора `#include`, що містить заголовочний файл `<stdio.h>` (`#include<stdio.h>`), головної функції `main()` (точка входу до програми) та її тіла у фігурних дужках, яке містить одну функцію. Програма завершується оператором `return 0`.

2. Наведіть приклад запису функції `main()` для простої програми?

```
#include<stdio.h>
int main() {
printf("Hello teacher!\n");
return 0;
}
```

3. Вкажіть етапи обробки коду для отримання файлу із розширенням `.exe`?

Після введення і редагування тексту програми на мові С створюється початковий файл з розширенням «.с». Наступний етап – це препроцесування, під час якого відбувається доповнення початкового файлу програми інформацією із заголовних файлів. Повний текст програми компілюється, у результаті чого утворюється об'єктні файли з розширенням «.о» або «.obj». Додаються бібліотечні функції, виконується компоунування, у результаті утворюється виконуваний файл з розширенням «.exe».

4. Що таке ідентифікатор? Що таке службові слова? Наведіть приклади службових слів. Чи можна використовувати службові слова в якості ідентифікаторів?

Ідентифікатор – ім'я об'єкта програми (змінної, масиву, структури, функції тощо), що дозволяє звернутись до об'єкта; ознака, яка цілком визначає сутність, в наперед визначеному просторі.

Службові слова – ряд зарезервованих ідентифікаторів, що вживаються для побудови конструкцій мови і мають фіксоване значення.

Службові слова не можна використовувати в якості ідентифікаторів, тому що це зарезервовані слова, імена яких не можуть обиратися довільним способом.

5. Назвіть правила складання ідентифікаторів? Наведіть приклади правильно створених ідентифікаторів?

При записі ідентифікаторів можуть використовуватися великі та малі літери латинської абетки, арабські цифри та символ підкреслення.

Ідентифікатор не може починатися з цифри і не може містити пробілів.

Компілятор мови C розглядає літери верхнього та нижнього регістрів як різні символи.

Приклади: КОМ_16, size88, _MIN, TIME, time.

6. Яким чином оголошуються константи в мові програмування C? Яка різниця між іменованими та неіменованими константами?

Константи можна оголосити за допомогою директиви препроцесора *#define*.

Коли оголошується іменована константа препроцесор пов'язує значення константи з її іменем.

Коли оголошується неіменована константа їй присвоюється тільки значення.

7. Назвіть базові типи даних: розмір в байтах, діапазон значень?

Тип	Розмір, байт	Значення
bool	1	true або false
unsigned short int	2	від 0 до 65 535
short int	2	від -32 768 до 32 767
unsigned long int	4	від 0 до 4 294 967 295
long int	4	від -2 147 483 648 до 2 147 483 647
int (16 розрядів)	2	від -32 768 до 32 767
int (32 розряди)	4	від -2 147 483 648 до 2 147 483 647
unsigned int (16 розрядів)	2	від 0 до 65 535
unsigned int (32 розряди)	4	від 0 до 4 294 967 295
char	1	від 0 до 255
float	4	від 1.2e-38 до 3.4e38
double	8	від 2.2e-308 до 1.8e308
long double	10	від 3.4e-4932 до 3.4e+4932

8. Який синтаксис оголошення та ініціалізації змінних?

Оголошення змінної відбудеться, якщо задати тип цієї змінної та зв'язати його з її ім'ям (наприклад, `int x`).

Ініціалізація змінної відбудеться, якщо присвоїти їй початкове значення (наприклад, `int x=1`).

9. Особливості застосування префіксної та постфіксної форм операції інкременту та декременту у виразах?

Під час застосування префіксної форми операції інкременту (декременту) спочатку збільшується (зменшується) значення змінної на одиницю, а потім обчислюється заданий вираз.

Під час застосування постфіксної форми операції інкременту (декременту) спочатку обчислюється заданий вираз, а потім збільшується (зменшується) значення змінної на одиницю.

10. Що таке специфікатор формату?

Специфікатори формату – це умовне позначення, яке дозволяє визначити тип даних для друку на стандартному виводі.