

Лабораторна робота № 12

Тема: C++. Адреси даних. Вказівники. Динамічна пам'ять.

Мета: Сформувати навички використання вказівників.

Зміст роботи

1. Нехай зроблені оголошення `int n=5, *Vk`. Які з наведених записів вірні?

Поясніть чому. Свою думку перевірте експериментальним шляхом.

- | | |
|---------------------------|-------------------------------|
| a. <code>Vk=&n</code> | d. <code>*Vk=n</code> |
| b. <code>Vk=n</code> | e. <code>*(&n)=Vk</code> |
| c. <code>Vk=*n</code> | f. <code>*Vk=*(&n)</code> |

1. Обчислити в байтах довжини вказівників на різні типи даних. Результат відобразити у звіті.
2. Напишіть програму виводу елементів довільного масиву використовуючи вказівники.
3. Створіть масив із перших 100 цілих чисел і обчисліть їх суму. Напишіть програму 2 способами з використанням індексу і вказівника.

Теоретичні відомості.

Перед використанням покажчики мають бути ініціалізовані. Як тільки покажчик одержав значення – адресу змінної, з'являється змога доступу до даних цієї змінної.

З покажчиками використовуються два оператори: "*" і "&". Оператор "&" – унарний. Він повертає адресу пам'яті, за якою розташований його операнд. Наприклад, у процесі виконання такого фрагмента коду програми

`ptr = &balance` у змінну `ptr` поміщається адреса змінної `balance`.

Ця адреса відповідає області у внутрішній пам'яті комп'ютера, яка належить змінній `balance`. Призначення оператора "&" можна "перекласти" українською мовою як "адреса змінної", перед якою він знаходиться. Щоб краще зрозуміти суть цього присвоєння, припустимо, що змінна `balance` розташована у області пам'яті з адресою 100. Отже, після виконання цього оператора змінна `ptr` набуде значення 100.

Другий оператор роботи з покажчиками (*) також унарний оператор, але він звертається до значення змінної, розташованої за адресою, заданою його операндом. Іншими словами, він посилається на значення змінної, яка адресується заданим покажчиком. Якщо змінна ptr містить адресу змінної balance, то у процесі виконання оператора `value = *ptr` змінній value буде присвоєне значення змінної balance, на яку вказує змінна ptr. Наприклад, якщо змінна balance містить значення 3200, то після виконання останнього оператора змінна value міститиме значення 3200, оскільки це те значення, яке зберігається за адресою 100.

Ілюстрація механізму використання покажчиків у поєднанні з операторами

```
int main()
{system ("cls"); // чистка екрану

setlocale(LC_ALL,"Ukrainian");
int balance;
int *ptr;
int value;
balance = 3200;
ptr = &balance;
value = *ptr;
cout << "Баланс дорівнює: " << value << endl;
Sleep(2000);
return 0;
}
Результат роботи програми:
Баланс дорівнює: 3200
```

Контрольні запитання

1. Як отримати адресу змінної?
2. Що таке вказівник?
3. Як здійснити перехід по адресі змінної?
4. В чому полягає відмінність динамічного масиву від стаціонарного?
5. Як оголосити динамічний масив?

4. Обчислити в байтах довжини вказівників на різні типи даних.
Результат відобразити у звіті.
5. Напишіть програму виводу елементів довільного масиву не використовуючи індекси.
6. Створіть масив із перших 100 цілих чисел і обчисліть їх суму.
Напишіть програму 2 способами з використанням індексу і вказівника.
7. Напишіть програму для рішення попередньої задачі використовуючи динамічний розподіл пам'яті.