

**Контрольні запитання**

1. **Які існують способи реалізації абстрактного типу даних «Список»?**

* Динамічний масив (Array-based)
* Зв’язний список (Node-based, Pointer-based)

1. **Якою є структура вузла у зв’язному списку?**

* значення (value), тобто власне дані, які зберігаються у вузлі
* вказівник (pointer, link), тобто посилання на сусідній елемент

Список завжди має голову списку (head) – перший елемент цього списку.

1. **Які є різновиди зв’язних списків?**

* Однозв’язний список (Singly Linked List)
* Двозв’язний список (Doubly Linked List)
* Кільцевий список (Circular Linked List)

1. **Яка реалізація абстрактного типу даних «Список» є найефективнішою?**

Переваги зв’язного представлення даних:

* можливість забезпечення значної змінності структур;
* розмір структури обмежується тільки доступним об’ємом машинної пам’яті;
* при зміні логічної послідовності елементів структури потрібно виконати не переміщення даних в пам’яті, а тільки корекцію покажчиків.

Разом з тим зв’язне представлення не позбавлене й недоліків, основні з яких:

* робота з покажчиками вимагає більш високої кваліфікації від програміста;
* на поля зв’язку витрачається додаткова пам’ять;
* доступ до елементів зв’язної структури може бути менш ефективним за часом

Зв'язані списки мають серію переваг порівняно з масивами. Зокрема, в них набагато ефективніше (за час О(1), тобто незалежно від кількості елементів) виконуються процедури додавання та вилучення елементів. Натомість, масиви набагато кращі в операціях, які потребують безпосереднього доступу до кожного елементу, що у випадку зі зв'язаними списками неможливо та потребує послідовного перебору усіх елементів, які передують даному.

**5. Як здійснюється пошук вузла у зв’язному списку?**

//Search an element in the list

Node temp = new Node();

temp = this.head;

found = 0;

i = 0;

if (temp != null)

{

while (temp != null)

{

i++;

if (temp.data == searchValue)

{

found++;

break;

}

temp = temp.next;

}

if (found == 1)

{

searchValue + " is found at index = " + i + ".";

}

}