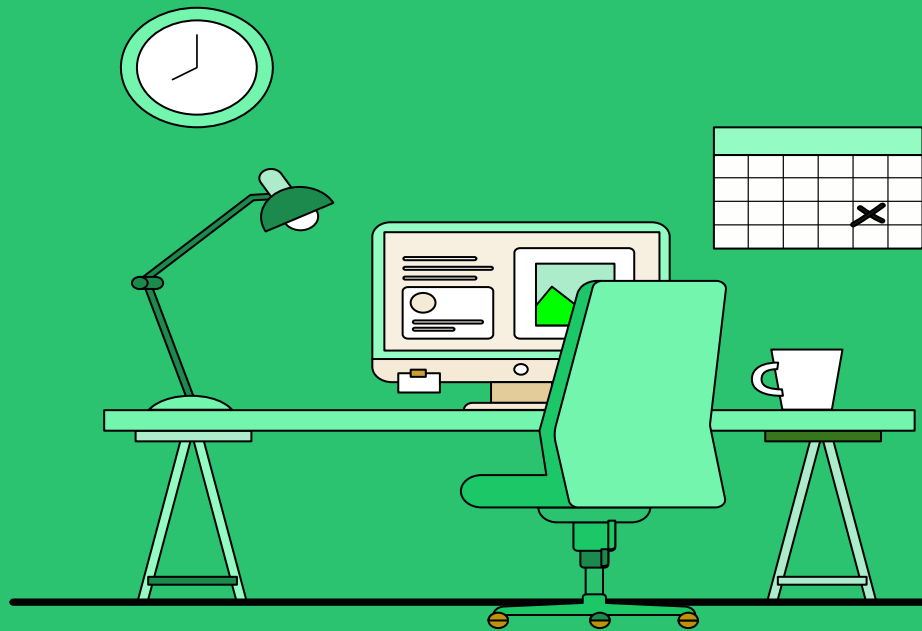


STL бібліотека

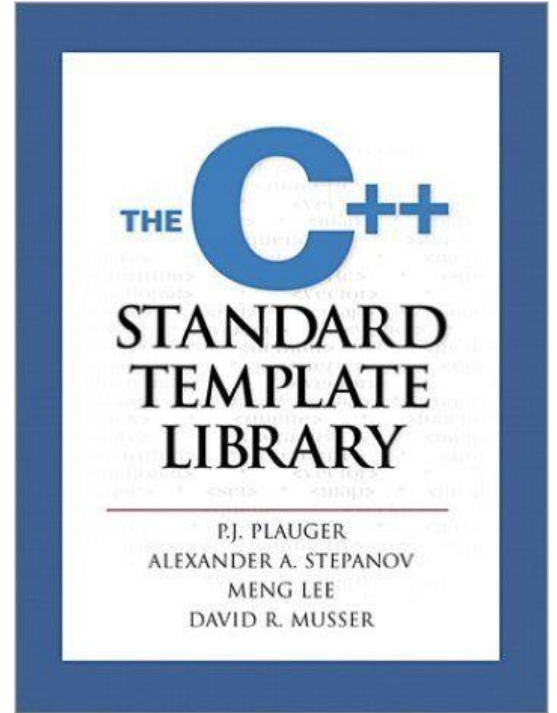
C++



Стандартна бібліотека шаблонів (STL) — це набір класів шаблонів C++ для забезпечення загальних структур даних та функцій програмування, таких як списки, стеки, масиви тощо. Це бібліотека класів контейнерів, алгоритмів та ітераторів.

Компоненти STL:

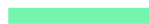
- Алгоритми
- Контейнери
- Функтори
- Ітератори



Колекції

Контейнер — це об'єкт, що зберігає колекцію інших об'єктів (його елементів).

Контейнер керує місцем для зберігання своїх елементів і надає функції-члени для доступу до них, безпосередньо або через ітератори (посилальні об'єкти з властивостями, подібними до вказівників).



Для використання колекції в кодї використовується директива `#include <T>`,

де `T` — назва колекції

vector — колекція елементів, збережених в масиві, що змінюється в міру необхідності розміру

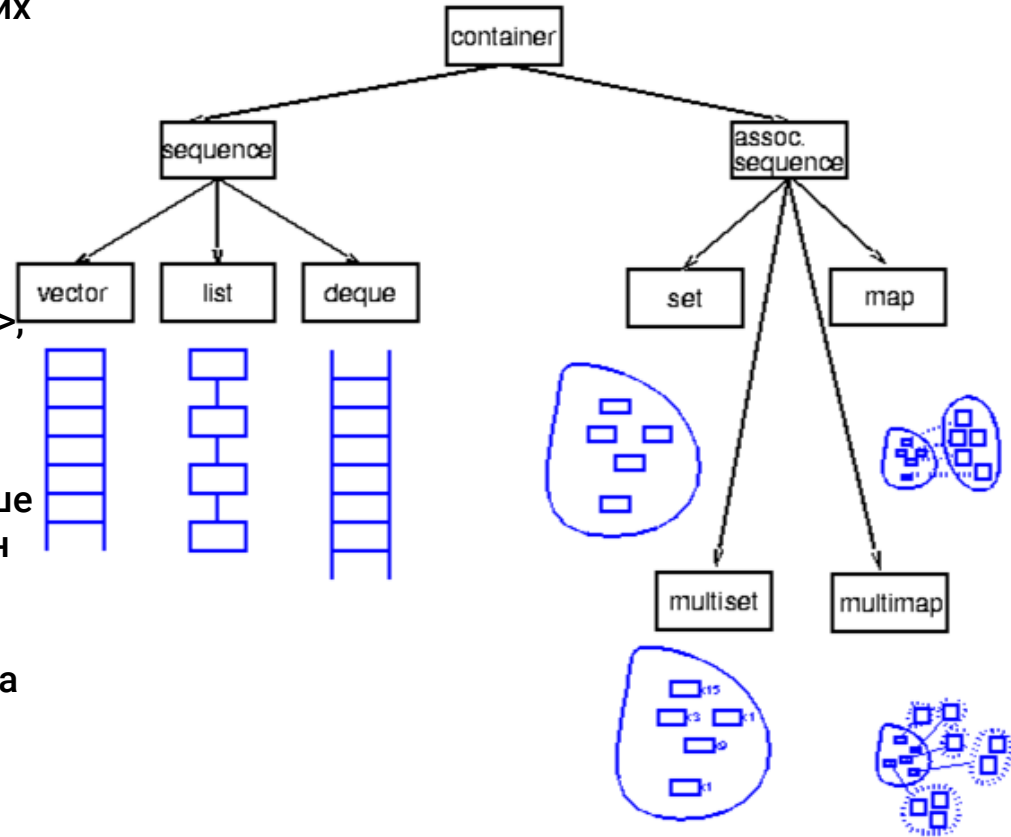
list — зберігає елементи в вигляді зв'язаного списку;

map — зберігає пари виду $\langle \text{const Key}, T \rangle$, тобто кожен елемент - це пара виду $\langle \text{ключ}, \text{значення} \rangle$

set — відсортована колекція одних лише значень, при цьому унікальних — кожен ключ може зустрітись тільки один раз

multimap — map, в якому відсутня умова унікальності ключа

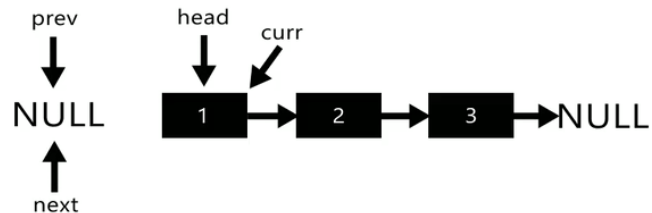
multiset — set з відсутністю умови унікальності ключа.



forward_list

Контейнер послідовностей, що дозволяють постійно виконувати операції вставки та стирання в будь-якому місці послідовності.

Однопов'язані списки можуть зберігати кожен з елементів, які вони містять, у різних і не пов'язаних між собою місцях зберігання.



```
while (current != NULL)
{
    next = current->next;
    current->next = prev;
    prev = current;
    current = next;
}
*head_ref = prev;
```

```
#include <iostream>
#include <forward_list>

int main()
{
    std::forward_list<int> numbers
    = { 1, 2, 3, 4, 5 };

    int first = numbers.front();

    for (int n : numbers)
        std::cout << n << "\t";
}
```

set

set — відсортована колекція одних тільки значень, при цьому унікальних — кожен ключ може зустрітися лише один раз.

```
#include <iostream>
#include <set>
#include <iterator>
using namespace std;
int main()
{
    set<char> mySet;
    mySet.insert('s');
    mySet.insert('e');
    mySet.insert('t');
    mySet.insert('t');
    copy( mySet.begin(), mySet.end(), ostream_iterator<char>(cout, " "));
    return 0;
}
```

Алгоритми

Алгоритм заголовка визначає набір функцій, спеціально розроблених для використання на діапазонах елементів. Вони діють на контейнери та забезпечують засоби для різних операцій щодо вмісту контейнерів.

Алгоритми

- Сортування
- Пошук
- Важливі алгоритми STL
- Корисні алгоритми масиву
- Операції розділів
- Числовий
- клас `valarray`

Ітератори

Ітератор — абстракція, яка забезпечує доступ до елементів колекції.

Типи ітераторів:

forward iterator — для обходу колекції від меншого індексу до більшого;
reverse iterator — для обходу колекції від більшого індексу до меншого;
random access iterator — для обходу колекції в будь-якому напрямку.

The C++ standard library generic containers

