

Лабораторна робота №11

Тема: Дослідження контейнерних класів бібліотеки STL.

Мета: дослідити контейнерні класи `vector` та `list` бібліотеки STL, набутти навичок їх використання.

Завдання:

1. З допомогою контейнера `std::vector` створити вектор значень типу `char`, в який записати український алфавіт (малими літерами).
2. Усі голосні букви у векторі замінити прописними (за допомогою циклу).
3. Вивести вміст вектора на екран.
4. З допомогою контейнера `list` створити список значень типу `char`, в який записати по буквах своє прізвище. Вивести вміст списку на екран.
5. Відсортувати список (від А до Я). У консоль вивести повідомлення: «Список сортується!».
6. Вивести вміст списку на екран.
7. Створити вектор об'єктів класу, створеного згідно індивідуального завдання №1 в лабораторній роботі №2. Заповнити вектор десятьма об'єктами.
8. Здійснити вивід значень об'єктів за допомогою індексу вектора.
9. Здійснити вивід значень об'єктів за допомогою ітераторів.
- 10.* З допомогою контейнера `list` створити список студентів групи. В контейнер поміщати об'єкти класу, в якому представлені такі атрибути: прізвище.

Лістинг:

Header.h:

```
#pragma once
#include <iostream>
#include <vector>
#include <list>
#include <string>

#define N 8

void setUaAlphabet(std::vector<char>& vec);
void setUpperCaseVol(std::vector<char>& vec);
void printVector(std::vector<char>&vec);
void fillListBySurname(std::list<char>&lt, std::string surname);
void printList(const std::list<char>& lt);

class Furniture {
private:
    std::string room;
    int weight;
public:
    Furniture();
    Furniture(std::string room, int weight);
    Furniture(const Furniture&);
```

```

    void setRoom(std::string room);
    std::string getRoom() const;
    void setWeight(int weight);
    int getWeight() const;
    void print() const;
    void input();
    ~Furniture();
};

class Student {
public:
    Student() : surname(""), name(""), fname(""), phoneNum(""), age(0) {}
    ~Student() {}
    void setSurname(std::string);
    void setName(std::string);
    void setFname(std::string);
    void setPhoneNum(std::string);
    void setAge(int);

    std::string getSurname() const;
    std::string getName() const;
    std::string getFname() const;
    std::string getPhoneNum() const;
    int getAge() const;

    friend std::ostream& operator <<(std::ostream& is, const Student st);
private:
    std::string surname;
    std::string name;
    std::string fname;
    int age;
    std::string phoneNum;
};

```

Source.cpp:

```

#include "Header.h"
using namespace std;

void Student::setSurname(string surn) {
    this->surname = surn;
}
void Student::setName(string name) {
    this->name = name;
}
void Student::setFname(string fname) {
    this->fname = fname;
}
void Student::setPhoneNum(string num) {
    this->phoneNum = num;
}
void Student::setAge(int age) {
    this->age = age;
}

string Student::getSurname() const {
    return this->surname;
}
string Student::getName() const {
    return this->name;
}
string Student::getFname() const {
    return this->fname;
}

```

```

}
string Student::getPhoneNum() const {
    return this->phoneNum;
}
int Student::getAge() const {
    return this->age;
}

ostream& operator <<(ostream& is, const Student st) {
    is << st.name << ", " << st.surname << ", " << st.fname << ", " << st.phoneNum <<
    ", " << st.age << endl;
    return is;
}

Furniture::Furniture() : weight(0), room("") {}
Furniture::Furniture(string room, int weight) {
    //cout << "Parametrized furniture constructor" << endl;
    setRoom(room);
    setWeight(weight);
}
Furniture::Furniture(const Furniture& src) {
    //cout << "Parametrized furniture constructor (link)" << endl;
    setRoom(src.getRoom());
    setWeight(src.getWeight());
}
Furniture::~Furniture() {}

void Furniture::setRoom(string room) {
    this->room = room;
}

void Furniture::setWeight(int weight) {
    this->weight = weight;
}
string Furniture::getRoom() const {
    return this->room;
}

int Furniture::getWeight() const {
    return this->weight;
}

void Furniture::input() {
    string nroom;
    int nweight;
    cout << "Enter room: ";
    cin >> nroom;
    setRoom(nroom);
    cout << "Enter weight of furniture: ";
    cin >> nweight;
    setWeight(nweight);
}

void Furniture::print() const {
    cout << "The furniture for room \"" << getRoom() << "\" has weight " <<
    getWeight() << endl;
}

void setUaAlphabet(vector<char>& vec) {
    int i = 224;
    while (i < 250) {
        switch (i) {
            case 228:
                vec.push_back(180);

```

```

        break;
    case 230:
        vec.push_back(186);
        break;
    case 233:
        vec.push_back(179);
        vec.push_back(191);
        break;
    }
    vec.push_back(i);
    i++;
}
vec.push_back(252);
vec.push_back(254);
vec.push_back(255);
}

void setUpperCaseVol(vector<char>& vec) {
    for (int i = 0; i < vec.size(); i++) {
        switch (vec[i]) {
            case 'a':
                vec[i] = 'A';
                break;
            case 'e':
                vec[i] = 'Y';
                break;
            case 'и':
                vec[i] = 'И';
                break;
            case 'i':
                vec[i] = 'I';
                break;
            case 'o':
                vec[i] = 'O';
                break;
            case 'y':
                vec[i] = 'Y';
                break;
        }
    }
}

void printVector(vector<char>& vec) {
    for (int i = 0; i < vec.size(); i++) {
        cout << vec[i] << " ";
    }
    cout << endl;
}

void fillListBySurname(list<char>& lt, string surname) {
    lt.empty();
    //for (list<char>::iterator it = lt.begin(), int i = 0; i < surname.size(); ++i) {
    for (int i = 0; i < surname.size(); ++i) {
        lt.push_back(surname[i]);
    }
}

void printList(const list<char>& lt) {
    lt.empty();
    //for (list<char>::iterator it = lt.begin(), int i = 0; i < surname.size(); ++i) {
    for (list<char>::const_iterator it = lt.cbegin(); it != lt.cend(); ++it) {
        cout << *it;
    }
    cout << endl;
}

```

lab11.cpp:

```
#include <iostream>
#include <vector>
#include <list>
#include <Windows.h>
#include "Header.h"

using namespace std;

int main()
{
    SetConsoleOutputCP(1251);
    SetConsoleCP(1251);

    vector<char> vec;
    vec.reserve(33);    // ukrainian alphabet has got 33 letters
    list<char> lst;

    cout << vec.capacity() << endl;
    setUaAlphabet(vec);
    printVector(vec);
    cout << vec.size() << endl;
    setUpperCaseVol(vec);
    printVector(vec);
    cout << vec.size() << endl;

    fillListBySurname(lst, "капусняк");
    printList(lst);
    cout << "Sorting..." << endl;
    lst.sort();
    printList(lst);

    vector<Furniture> furn;
    furn.reserve(10);
    for (int i = 0; i < 10; ++i) {
        furn.push_back(Furniture("kitchen", i * 10 + 1));
    }

    for (int i = 0; i < furn.size(); ++i) {
        furn[i].print();
    }
    cout << endl;
    for (vector<Furniture>::iterator it = furn.begin(); it != furn.end(); ++it) {
        it->print();
    }

    list<Student> stud;
    Student st;
    st.setName("Володимир");
    st.setSurname("Великий");
    st.setFname("Батькович");
    st.setPhoneNum("+380950670680");
    st.setAge(28);

    stud.push_back(st);

    st.setName("Андрій");
    st.setSurname("Стрий");
    st.setFname("Васильович");
    st.setPhoneNum("+380676789003");
    st.setAge(28);
```

```

stud.push_back(st);

cout << endl;
for (list<Student>::iterator it = stud.begin(); it != stud.end(); ++it) {
    cout << *it;
}

return 0;
}

```

Результат:

```

33
а б в г г д е є ж з и і ї й к л м н о п р с т у ф х ц ч ш щ ь ю я
33
А Б В Г Г Д У Є Ж З И І І Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ ь Ю Я
33
капусняк
Sorting...
аккнпсуя
The furniture for room "kitchen" has weight 1
The furniture for room "kitchen" has weight 11
The furniture for room "kitchen" has weight 21
The furniture for room "kitchen" has weight 31
The furniture for room "kitchen" has weight 41
The furniture for room "kitchen" has weight 51
The furniture for room "kitchen" has weight 61
The furniture for room "kitchen" has weight 71
The furniture for room "kitchen" has weight 81
The furniture for room "kitchen" has weight 91

The furniture for room "kitchen" has weight 1
The furniture for room "kitchen" has weight 11
The furniture for room "kitchen" has weight 21
The furniture for room "kitchen" has weight 31
The furniture for room "kitchen" has weight 41
The furniture for room "kitchen" has weight 51
The furniture for room "kitchen" has weight 61
The furniture for room "kitchen" has weight 71
The furniture for room "kitchen" has weight 81
The furniture for room "kitchen" has weight 91

Володимир, Великий, Батькович, +380950670680, 28
Андрій, Стрий, Васильович, +380676789003, 28

```

Висновок: дослідив контейнерні класи vector та list бібліотеки STL, набув навичок їх використання.