

## Стандардна библиотека шаблони. Контејнери и нивна поделба – час 3

Стандардната библиотека на шаблони (**STL - Standard Template Library**) претставува множество од структури и алгоритми кои може да вршат операции над различни типови на податоци.

Својствата на **стандардната библиотека** се сместени во просторот **std**. Стандардната библиотека може да се подели на два дела:

- Стандардна библиотека на функции (The Standard Function Library): каде се сместени функциите за општа намена и не се дел од класа
- Библиотека на ОО класи (The Object Oriented Class Library): е множество на класи и соодветни членки на функции

**Стандардна библиотека на функции** ги има следниве категории:

- Влезно / излезни функции (I/O)
- Функции за работа со стрингови и знаци
- Математички функции
- Функции за работа со датум и време
- Функции за управување со меморијата
- Други функции

**Библиотека на ОО класи** дефинира дополнително множество класи кои овозможуваат поддршка на многу секојдневни активности како влез/излез, работа со стрингови и обароботка на броеви. Оваа библиотека е составена од:

- Стандардни класи за влез / излез
- Класи за работа со стрингови
- Класи за работа со броеви
- STL контејнерски класи
- STL алгоритми
- STL функциски објекти
- STL итератори
- STL алокатори
- Класи за управување со исклучоци
- Библиотека за локации
- Библиотека за разни поддршки

**Контејнерите** претставуваат структури на податоци во кои може да се чуваат разни информации. Досега, во сите предавања, зборувавме за само еден симплифициран "контејнер" на податоци: низата. Всушност, во целиот програмски јазик C++ (без STL), не постои друг контејнер во кој може да се сместуваат повеќе податоци од еден ист тип. Сите контејнери кои се дел од стандардната библиотека на шаблони се имплементирани како параметаризирани структури. Ова значи дека еден контејнер може да служи за чување на податоци од кој било тип (int, float, double, string, vector<int>, vector<float>, vector<vector<float> >, итн). За да може да користиме еден контејнер за чување на податоци, потребно е да ја вклучиме датотеката во која е тој дефиниран, тоа го правиме пр. со наредбата `#include <vector>`.

Стандардната библиотека (C++ Standard Library) е множество од класи и функции напишани во C++. Обезбедува неколку **генерички контејнери**, функции за работа со контејнерите, објекти на функциите, генерички функции (како стандардни функции за влез и излез, работа со датотеки) и стандардни математички функции.

**Контејнери** (садови) во секојдневниот живот се на пример страниците на книгата во кориците на книгата, старите предмети сместени во кутии и слично. Слично на ова, контејнерската класа е создадена за во неа да се сместуваат и организираат повеќе инстанци на други класи. Постојат многу различни контејнерски класи, но основна е стандардната библиотека на шаблони (STL - Standard Template Library). Овозможува користење на контејнерски објекти како: вектори, листи, множества, мапи и др. Контејнерските класи се групирани во три категории:

- **секвенционални** (вектори, ред со два краја, листи, низи, насочени листи) – кај овој вид контејнери секој елемент има своја позиција. Позицијата зависи од времето и местото на вметнување. Ако се додадат четири елементи на крајот на векторот, елементите ќе се појават во точен редослед како што се вметнати. Обично се имплементираат со полиња и поврзани листи.

- **асоцијативни** (множество, мапа, мултимапа) – овде позицијата на елементот зависи од неговата вредност. Вредноста на елементот ја одредува позицијата на елементот во контејнерот. Редоследот на вметнување не е важен. Обично се имплементираат со бинарни дрва.

- **адаптивни (неограничени)** контејнери (стек и ред, неподредено множество, неподредена мапа) овде не е важна ни позицијата на елементот, не редоследот на внесување ниту пак вредноста на елементот. Важно е само нивното постоење во контејнерот. Ако се стават четири елементи, нивниот редослед не е дефиниран и може да се смени било кога. Обично се имплементираат со хеш табели.

STL има и голем број генерички **алгоритми** за работа со контејнерските класи. Овие алгоритми овозможуваат сортирање, пребарување, додавање, промена на редоследот, бришење и копирање на елементи на класата.

**Итераторите** претставуваат еден вид покажувачи кон елементите од контејнерите. Итераторите се користат за движење низ елементите од еден контејнер, како и за означување на делови од податоци врз кои сакаме да вршиме операции.

*Прашања поврзани со наставните единици може да се испраќаат на email:*

[anetastojceska@gmail.com](mailto:anetastojceska@gmail.com)