

Пројектни задатак обухвата пројектовање и имплементацију шаблона класе која представља базен објеката одређеног предефинисаног типа. У програмима који захтевају честу алокацију (најчешће дубоких типова) података оваква класа се користи као један од начина за постизање боље перформансе извршавања јер редукује време потребно за заузимање нове меморије и конструисање објеката у њој.

Захтеви:

- Класа током конструисања заузима у меморији (`new`) одређени број објеката дефинисаног типа (параметар шаблона).
- Класа дефинише јавну функцију чланицу `acquireObject()` помоћу које се преузимају слободни објекти. Један позив функције - један објекат предефинисаног типа.
- Ако у моменту позива функције `acquireObject()` у базену нема слободних објеката, онда се заузимају нови, тј. базен се проширује.
- Функција `acquireObject()` треба да врати `shared_ptr<T>`.
- Треба обезбедити да се објекат не обрише када нестане и последњи дељени показивач на њега, већ да се врати у базен незаузетих објеката. То се може постићи додавањем посебне функције која ће се позвати при уништавању последњег дељеног показивача. За више погледати конструктор (4) на следећој страници: https://en.cppreference.com/w/cpp/memory/shared_ptr/shared_ptr (Links to an external site.).
- За разлику од `new`, `acquireObject()` не гарантује конструкцију објекта. Сваки објекат из базена ће бити конструисан само једном, тада када буде направљен и смештен у базен, тј. кад је меморија за њега заузета са `new`. Такође, деструктор објекта се не позива када се он враћа у базен. Деструктор сваког објекта из базена ће бити позван само на крају, из деструктора базена.

Потребно је написати једноставан пример програма који ће демонстрирати развијену функционалност.