
Programowanie Obiektowe w C++

LISTA 1 (40 PKT)

DEADLINE'Y DLA GRUP: 3, 7, 8.01.2013

Zaimplementuj metodą wskaźnikową klasę **STOS** składającą się z obiektów klasy **ELEMENT**. Klasa **ELEMENT** reprezentuje elementy stosu i powinna przechowywać łańcuch znaków oraz wskaźnik na poprzedni element stosu. Więcej o stosie znajdziesz tutaj:

- [http://pl.wikipedia.org/wiki/Stos_\(informatyka\)](http://pl.wikipedia.org/wiki/Stos_(informatyka)) – wersja podstawowa
- <http://www.cmpe.boun.edu.tr/~akin/cmpe223/chap2.htm> – aplet ilustrujący działanie stosu podlinkowany na dole strony

ZADANIA:

1. (8 pkt) Zdefiniuj klasy **STOS** i **ELEMENT**. Pamiętaj o poprawnym określeniu stałych (const-correctness) oraz prywatnych składowych klas.
2. (4 pkt) Dla klasy **STOS** zdefiniuj konstruktor domyślny, kopiujący, operator przypisania oraz destruktor. Jeżeli w Twojej implementacji istnieje taka potrzeba, zdefiniuj te elementy także dla klasy **ELEMENT**. Pamiętaj o konstruktorze klasy **ELEMENT**, który umożliwi utworzenie obiektu z łańcucha znaków.
3. (4 pkt) W klasie **STOS** zdefiniuj metody dostępu, wstawiania oraz usuwania elementu stosu – pamiętaj, że do stosu dostęp jest tylko z jednej strony.
4. (4 pkt) W klasie **STOS** zdefiniuj metody: czyszczącą stos, zwracającą aktualny rozmiar oraz sprawdzającą, czy nie jest pusty. W klasie **STOS** zdefiniuj metodę zwracającą wskaźnik na element stosu.
5. (4 pkt) Twoja implementacja klasy **STOS** powinna być bezpieczna, tzn. chronić przed wyciekami pamięci i zapewniać poprawność obiektu w razie wystąpienia wyjątku w trakcie wykonywania funkcji. Zastosuj technikę znaną jako *zajmowanie zasobów jest inicjacją* (Resource Aquisition is Initialization), w razie potrzeby użyj bloków try-catch z obsługą błędów w funkcjach składowych lub wykorzystaj sprytny wskaźnik `unique_ptr` z biblioteki standardowej.
6. (6 pkt) Zapoznaj się z pojęciem *odwrotna notacja polska* (szczegóły: http://pl.wikipedia.org/wiki/Odwrotna_notacja_polska). Wykorzystaj klasy **STOS** i **ELEMENT** do konwersji łańcucha znaków z notacji infiksowej do ONP.
7. (5 pkt) Wykorzystaj klasy **STOS** i **ELEMENT** do obliczenia wartości wyrażenia ONP.
8. (5 pkt) Wykorzystaj klasy **STOS** i **ELEMENT** do wczytania pliku tekstowego, który w każdej linii ma inne wyrażenie matematyczne w postaci infiksowej. Zamień każde wyrażenie na postać ONP i oblicz wartość. Wynik (wyrażenie w ONP i wartość) wypisz na ekran oraz zapisz do pliku.