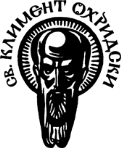
**СУ “Св. Климент Охридски”,**

**Факултет по математика и информатика ,Софтуерно инженерство**

**Курсов проект по Обектно-ориентирано програмиране**

**Двусвързан списък**

***Деян Даниелов Денчев, № 61820***

****

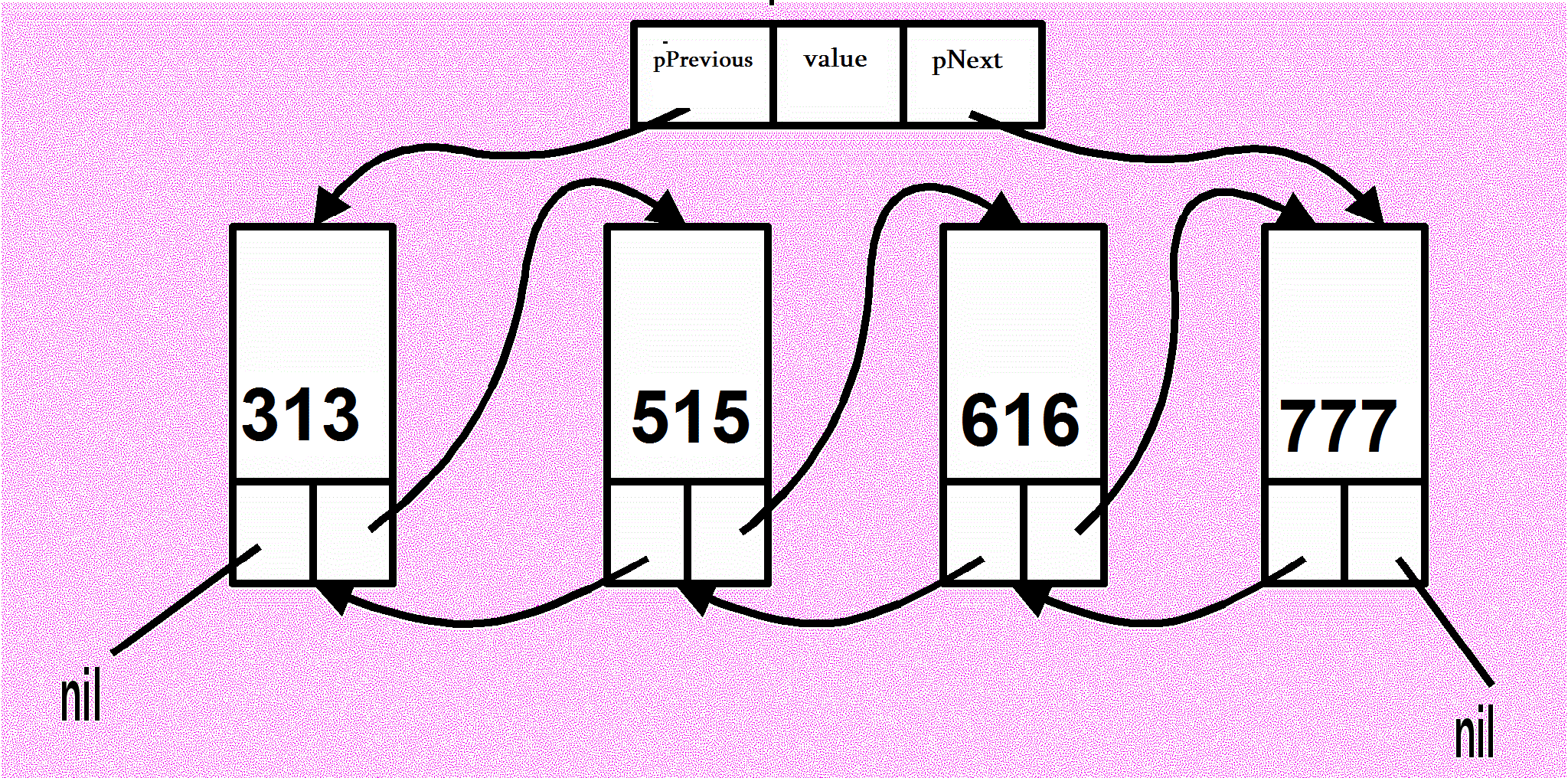
***Съдържание***

[1. Въведение 3](#_Toc421456369)

[2.Приложени алгоритми 3](#_Toc421456370)

[3. Програмен код 4](#_Toc421456371)

[4. Технологии 5](#_Toc421456372)

****

# ***D:\Documents\СУ\Домашни, контролни, изпити\ОС\Проект\emblema.pngВъведение***

Проектът представлява двусвързан списък -линейна структура от свързани еднотипни елементи. За реализрацията му е използван шаблонен клас. Отделните елементи са референтни променливи от тип запис с три полета:

1. *Поле със стойност*
2. *Поле сочещо към предходния елемент от списъка*
3. *Поле сочещо към следващия елемент от списъка*

Включените функционалности в двусвързания списък са:

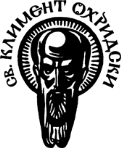
* добавяне на елемент в началото на списъка
* премахване на елемент в началото на списъка
* добаване на елемент в края на списъка
* премахване на елемент в края на списъка
* добаване и премахване на елемент на произволна позиция в списъка
* сравняване на записите в списъка
* изтриване на всички елементи от списъка
* проверка на големината на списъка
* проверка дали списъкът е празен.

**\*** За удобство е създадена и допълнителна променлива - брояч на елементите в списъка

# ***Приложени алгоритми***

Използваните алгоритми в задачата са за добавяне на елемент – ***push*\_*front*, *push*\_*back*,** ***insert***  и за премахване на елемент – ***pop*\_*front*, *pop\_back*, *erase*.**

Методите за добавяне и премахване не елемент в началото и края на списъка работят на следния принцип:

1. ****Проверка дали списъка е празен. Ако е празен,при добавяне на елемент, го прави първи и последен, за да е свързан списъка.
2. Ако списъкът не е празен, в зависимост от посоката, в която се изтрива или добавя елемент, се прави връзка с предходния/следващия елемент, като връзката е двустранна – свързват се указателите ***pPrevious* и *pNext*.**
3. Също така се увеличава и размерът на списъка.

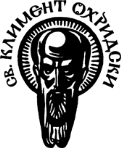
За добавяне/премахване на елемент на определена позиция, командите (***insert* *и* *erase*)** работят чрез обхождане с ***iterator*.**При добавяне в началото или в края на списъка, алгоритъмът е като този при ***push*\_*back*, *push*\_*front,* *pop*\_*back* и *pop*\_*front***. При добавянето или премахването на елемент от произволна позиция **iterator**,чрез нов временен ***Node*** се вмъква конкретната стойност и се свързва със ***Node***-овете от двете му страни.

# ***Програмен код***

Проектът се състои от три класа :***Node***, ***List*** – списък от указатели към Node (***Node*\*), *Iterator*** (който е имплементиран вътре в класа ***List*)**. ***Iterator*** служи за обхождане на списъка,а класът ***Node*** представлява отделните елементи на списъка.

Класът Node съдържа три член-данни:

* стойност на елемента
* указател към предишния
* указател към следващия ***Node***.

**** В този клас са имплементирани сетъри и гетъри за лесен достъп и промяна на стойностите на свързаните с текущия ***Node*** елементи:

* ***getVal***
* ***getPrevious***
* ***getNext***
* ***SetPrevious***
* ***SetNext***

Шаблонният клас ***List***  представлява двусвързания списък от обекти от тип ***Node***. Той има за член данни указател към първия и последния елемент на списъка и размер на списъка.

Класът ***List*** съдържа конструктор по подразбиране(създава празен списък), copy конструктор(създава списък от друг, вече съществуващ) и деструктор. Класът включва и функции ***front*** – достъп до елемента в началото на списъка, ***end*** – достъп до елемента в края на списъка, ***get*\_*size*** – връща размерността на списъка***, empty*** – проверява дали списъкът е празен. и ***clear*** – изтрива елементите на списъка. В класа ***Node*** е включен и клас **iterator**, имащ като член данна указател към текущ ***Node***. Чрез методите ***getCurrent*, *setCurrent*** и ***предефинирани* оператори**, класът реализира обхождането на листа.

# ***Технологии***

Проектът е реализиран чрез езика C++,а използваната виртуална среда е ***Microsoft Visual Studio 2013.***