Санкт-Петербургский государственный электротехнический университет

“ЛЭТИ”

кафедра МО ЭВМ

**ОТЧЕТ**

**по практической работе № 1**

**«Обработка однонаправленных списков»**

Выполнил Петухов С.В.

Факультет КТИ

Группа № 2304

Преподаватель

“Выполнено” “ ” \_\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_

2014 г.

1. Постановка задачи

Реализовать односвязный список на ЯВУ Java

1. Вывод

Программа исполняет определённый сценарий: список заполняется символами алфавита, содержимое списка выводится в консольном режиме, список очищается, содержимое списка выводится снова для проверки пустоты.

1. Описание структур данных:

|  |  |  |
| --- | --- | --- |
| Класс Link | | |
| Имя переменной | Тип переменной | Назначение |
| character | char | Значение ячейки(символ) |
| next | Link | Указатель на следующий элемент |

|  |  |  |
| --- | --- | --- |
| Класс LinkList | | |
| Имя переменной | Тип переменной | Назначение |
| first | Link | Указатель на голову |

Класс Link

|  |  |  |
| --- | --- | --- |
| Имя функции | Тип функции | Назначение |
| Link |  | Конструктор |
| displayLink | void | Вывод значения ячейки списка |

Класс LinkList

|  |  |  |
| --- | --- | --- |
| Имя функции | Тип функции | Назначение |
| LinkList |  | Конструктор |
| isEmpty | boolean | Проверка на пустоту списка |
| insertFirst | void | Вставить ячейку |
| deleteFirst | Link | Удалить ячейку |
| displayList | void | Вывод содержимого списка |

1. Листинг программы

link.java

**class** Link

{

**public** **char** character;

**public** Link next;

**public** Link(**char** character)

{

**this**.character = character;

}

**public** **void** displayLink()

{

System.*out*.print(character + " ");

}

}

**class** LinkList

{

**private** Link first;

**public** LinkList()

{

**this**.first = **null**;

}

**public** **boolean** isEmpty()

{

**return** (first == **null**);

}

**public** **void** insertFirst(**char** character)

{

Link newLink = **new** Link(character);

newLink.next = first;

first = newLink;

}

**public** Link deleteFirst()

{

Link temp = first;

first = first.next;

**return** temp;

}

**public** **void** displayList()

{

System.*out*.println("\nLinkedList: ");

Link temp = first;

**while**(temp != **null**)

{

temp.displayLink();

temp = temp.next;

}

System.*out*.println();

}

}

linkListProgram.java

**public** **class** LinkListProgram

{

**public** **static** **void** main(String[] args)

{

LinkList linkList = **new** LinkList();

**for** (**int** i = 0; i < 26; ++i)

linkList.insertFirst((**char**)(65 + i));

linkList.displayList();

**while**(!linkList.isEmpty())

{

linkList.deleteFirst();

}

linkList.displayList();

}

}