**Загвар класс**

Лаб №10

Д.Самдан (16B1SEAS2873)

ХШУИС, ПХ-ийн 3-р түвшин, pri.samdan@gmail.com

**1. ОРШИЛ**

Энгийн классын гишүүн өгөгдөл, функцийн параметр, буцаах утга нь статик байдаг бол загвар класс ашиглсанаар гишүүн өгөгдөл, функцийн параметр, буцаах утга нь ашиглах өгөгдөлийн төрөлөөс хамааран өөр өөр болдог. Энэ ойлголтыг ашиглан ямар ч төрлийн өгөгдлийг хадгалах жагсаалт (LinkedList) үүсгэсэн.

**2. ЗОРИЛГО**

Загвар классыг хэрхэн Java хэлэнд үүсгэх талаар судлах. LinkedList-н талаар судлах. Загвар класс ашиглах LinkedList үүсгэх. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Загвар классын талаар судлах.
2. LinkedList-ийн талаар судлах.
3. Загвар класс ашиглах LinkedList үүсгэх.
   1. Загвар класс ашиглан элемент класс байгуулах
   2. Гишүүн өгөгдлийн үүсгэх.
   3. Гишүүн функцийг тодорхойлох

**3. ОНОЛЫН СУДАЛГАА**

**Загвар – Generics in Java**

Genrics нь С++ хэлний загвартай төстэй ойлголт юм. Энэ нь функц, класс, интерфейст олон төрлийн өгөгдлийг параметрээр авах боломжийг олгодог[1].

- **Загвар класс**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | **public** **class** **Test**<T> {  **private** T a;  **public** T **getA**() {  **return** a;  }  **public** **void** **setA**(T a) {  **this**.a = a;  }  } |

Классын нэрийн хойно <T> гэж бичнэ. T өгөгдлийн төрлийг тухай нь классад ашиглан гэсэн үг юм. Энэ T нь объектоос үүсгэх үед авах төрөл ба тухайн авсан төрлөөс хамааран a гишүүн өгөгдөл, функцийн авах, буцаах төрөл өөр өөр байж болно.

- Загвар классын объектын байгуулах

|  |  |
| --- | --- |
| 1  2 | Test<Integer> a = **new** Test<Integer>();  Test<Float> b = **new** Test<Float>(); |

Загвар классын заалт үүсгэхэд классын нэрийн хойно <> дотор төрөлийн хамт дамжуулна.

- **Загвар функц**

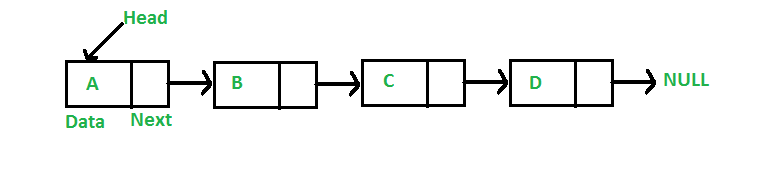
Загвар функц гэдэг нь функц параметтээ дурын өгөгдөлийн төрөл авах функцийг хэлнэ. Загвар функцийг үүсгэхэд функцийн буцаах утгын өмнө <E> гэж бичнэ[1].

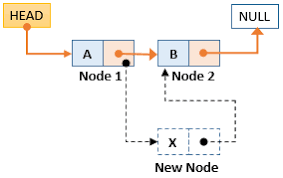
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | **public** **class** **Example** {  **public** **static** < E > **void** printArray( E[] inputArray ) {  // Display array elements  **for**(E element : inputArray) {  System.out.printf("%s ", element);  }  System.out.println();  }  **public** **static** **void** **main**(String[] args) {  Integer[] intArray = { **1**, **2**, **3**, **4**, **5** };  String[] stringArray = {"a", "b", "c", "d", "e"};  printArray(intArray);  printArray(stringArray);  }  } |

Дээрх код нь printArray гэсэн загвар функцийг тодорхойлсон ба энэ нь өгөгдсөн хүснэгтийг хэвлэх функц юм. Түүнийгээ ашиглан бүхэл тоо болон тэмдэгт мөрүүдийг хэвлэж байна.

**LinkedList**

LinkedList нь хүснэгт шиг олон элемент хадгалахад ашиглагдаг өгөдлийн бүтэц юм, гэвч хүснэгт шиг элемент нь дис дараалан орших албагүй. Элемент бүрд өөрийн утга, дараагын элементийн байрлалыг заана[2].

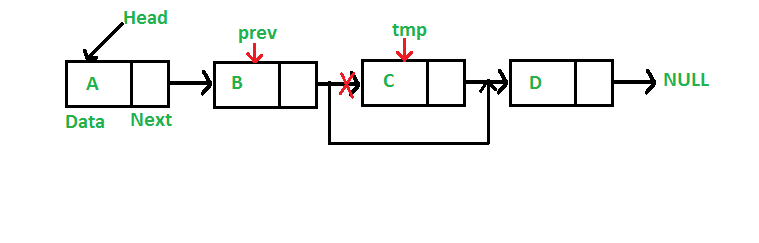


**LinkedList-д элемент нэмэх**

Элемент нэмэх гэж буй газрын өмнөх элементийн дараагын элемент зааж буй заалтыг нь шинэ элементийн дараагын элемент рүү заах заалтад өгөөд шинэ элементийг өмнөх элементийн дараагын элементийн заалтад өгнө.

**LinkedList-аас элемент хасах**

Хасах гэж буй элементийн өмнөх элементийн дараагын элемент заах заалтад нь хасаж буй элементийн дараагын элементийн заах заалтын өгөөд. Хасах гэж буй элементийг заах заалт үгүй болсон тул тухайн элемент устан.



**4.Хэрэгжүүлэлт**

**Node.java** – Энэ класс нь LinekdList-ийн нэг элемент ба дурын өгөгдлийн төрөл хадах боломжтой загвар класс юм. Энэ класс нь өөрийн утга хадгалах value, дараагын элементийг заах next\_node гэсэн хоёр гишүүн өгөгдөлтэй байна.

**LinkedList.java** – Энэ класс нь дурыг өгөгдлийн төрлийг LinkedList өгөгдлийн бүтэц ашиглан хадгалах загвар класс юм. Энэ класс нь хамгийн эхний элемент буюу цувааны толгойг агуулах ба түүнийгээ ашиглан элемент нэмэх, хайх, хасах, уртын нь олон функцүүдай байна.

**5. ДҮГНЭЛТ**

Загвар классыг ашигласнаар олон өгөгдлийн төрөлт ажиллах нэгэн классыг бий болгож чадна. Үүнийг ашиглан бүх өгөгдлийн төрөлд ажиллах жагсаалт(LinkedList) классыг байгуулсан.

**Номзүй**

1. Java - Generics, <https://www.tutorialspoint.com/java/java_generics.htm>
2. Linked List | Set 1 (Introduction), <https://www.geeksforgeeks.org/linked-list-set-1-introduction/>

**ХАВСРАЛТ**

**Node.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | **public** **class** **Node**<T> {  T value;  Node next\_node;  **public** **Node**(T value) {  **this**.value = value;  **this**.next\_node = **null**;  }  **public** **Node**(T value, Node<T> temp) {  **this**.value = value;  **this**.next\_node = temp;  }  // GET  **public** T **getValue**() {  **return** value;  }  **public** Node **getNext\_node**() {  **return** next\_node;  }  // SET  **public** **void** **setValue**(T value) {  **this**.value = value;  }  **public** **void** **setNext\_node**(Node<T> next\_node) {  **this**.next\_node = next\_node;  }  } |

**LinkedList.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107 | **public** **class** **LinkedList**<T> {  Node<T> head;  **int** length;  **public** **LinkedList**() {  **this**.length = **0**;  }  **public** **LinkedList**(T value) {  head = **new** Node<T>(value);  **this**.length = **1**;  }  **public** **void** **add**(T value) {  add(**this**.head, value);  **this**.length++;  }  **private** Node<T> **add**(Node<T> current, T value) {  **if** (current == **null**) {  current = **new** Node<T>(value);  } **else** {  current.next\_node = add(current.next\_node, value);  }  **return** current;  }  **public** **void** **insert**(T value, **int** index) {  **if** (index < **this**.length - **1**) {  **this**.head = insert(**this**.head, value, index);  **this**.length++;  } **else** {  add(value);  }  }  **private** Node<T> **insert**(Node<T> current, T value, **int** index) {  **if** (index == **0**) {  Node<T> temp = current;  current = **new** Node<T>(value, temp);  } **else** {  current.next\_node = insert(current.next\_node, value, --index);  }  **return** current;  }  **public** T **get**(**int** index) {  **if** (index > **this**.length - **1**)  **return** **null**;  Node <T> current = **this**.head;  **while** (index != **0**) {  current = current.next\_node;  index--;  }  **return** current.value;  }  **public** **int** **length**() {  **return** length;  }  **public** **int** **length\_rec**() {  **return** **length\_rec**(**this**.head);  }  **void** **delete**(**int** index) {  **if** (index > **this**.length)  index = **this**.length - **1**;  **this**.length--;  **if** (index == **0**) {  Node temp = **this**.head;  **this**.head = temp.next\_node;  temp = **null**;  **return**;  }  delete(**this**.head, index);  }  **private** Node<T> **delete**(Node<T> current, **int** index) {  **if** (index == **1**) {  Node <T> temp = current.next\_node;  current.next\_node = temp.next\_node;  temp = **null**;  } **else** {  current.next\_node = delete(current.next\_node, --index);  }  **return** current;  }  **private** **int** **length\_rec**(Node current) {  **if**(current != **null**) {  **return** **1** + length\_rec(current.next\_node);  } **else** {  **return** **0**;  }  }  } |

**Main.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | **public** **class** **Main** {  /\*  Template буюу загвар класс ашиглан дурын төрлийн  өгөгдөл хадгалдаг жагсаалтыг зохион байгуул.  ПХ, КУ, МТ ангийн хүүхдүүд linked list зохион байгуулна.  Жагсаалт нь дараах функцуудтай байна:  \* void add(T t); // жагсаалтын сүүлд элемент нэмнэ  \* void insert(T t, int index); // index - дүгээрт элеменет оруулна  \* T get(int index); // i-р элементийн утгыг буцаана  \* void delete(int index); // i -р элементийг устгана  \* int length(); //жагсаалтын уртыг буцаана  \*/  **public** **static** **void** **main**(String[] args) {  LinkedList<Integer> a =**new** LinkedList<Integer>(**10**);  a.add(**20**);  a.add(**30**);  a.add(**40**);  a.delete(**9**);  // a.insert(6, 0);  System.out.println(a.get(a.length - **1**));  System.out.println(a.length\_rec());  }  } |