## 3.2 Алгоритмы по шагам

Алгоритм функции**- void insert(const Iterator &past, const T &past\_data)** класса **List<T>.** Является функцией шаблонного класса, предназначена для вставки элемента с данными past\_data в любое место в списке.

Шаг 1: Начало.

Шаг 2: Объявляется переменная **flag** типа bool и инициализируется значением 1.

Шаг 3: Создаем объект **temp** класса Iterator при помощи конструктора по умолчанию и инициализируем результатом вызова функции this->begin().

Шаг 3: Цикл while по переменной temp c выходным условием temp!= this->end().

Шаг 4: Если временная переменная итератора temp равна переменной past, предполагаемому месту вставки элемента, то перейти к шагу.

Шаг 5: Конец цикла while по переменной temp.

Шаг 6: Если временная переменная итератора temp равна переменной past, предполагаемому месту вставки элемента, то перейти к шагу 16.

Шаг 7: Создаем указатель **temp** на объект класса Cell<T> и выделяем память под 1 элемент класса Cell<T>.

Шаг 8: Создаем объект **it** класса Iterator при помощи конструктора по умолчанию и инициализируем переменной past класса Iterator, указывающей на место добавления нового элемента.

Шаг 9: Инициализируем поле data объекта temp данными вставляемого элемента из переменной past\_data.

Шаг 10: Инициализируем указатель на следующий элемент объекта temp адресом переменной it.

Шаг 11: Инициализируем указатель на предыдущий элемент объекта temp адресом указателя на предыдущий элемент объекта it.

Шаг 12: Присваиваем указателю на предыдущий элемент объекта it адрес указателя на предыдущий элемент объекта temp.

Шаг 13: Если указатель на предыдущий элемент объекта temp не равен nullptr, то переход к шагу 17.

Шаг 14: Если адрес it равен адресу предыдущего элемента объекта this, то переход к шагу 18 .

Шаг 15: Если адрес it равен адресу первого элемента объекта this, то переход к шагу 19 .

Шаг 16: Присваиваем flag значение 0 и переходим к шагу 5.

Шаг 17: Присваиваем temp->previous->next адрес объекта temp.

Шаг 18: Присваиваем указателю на последний элемент объекта адрес указателя на следующий элемент объекта temp.

Шаг 19: Присваиваем указателю на первый элемент объекта адрес this адрес указателя на предыдущий элемент &it.

Шаг 20: Увеличиваем размер списка на единицу.

Шаг 20: Выход из функции.

- **void push\_back(T to\_add)**

Является функцией шаблонного класса **List<T>**, предназначена для добавление элемента в конец списка.

Шаг 1: Начало.

Шаг 2: Создаем указатель **temp** на объект класса Cell<T> и выделяем память под 1 элемент класса Cell<T>..

Шаг 3: Инициализируем поле data объекта переменной to\_add.

Шаг 4: Инициализируем указатель на следующий элемент объекта temp nullptr.

Шаг 5:Если размер списка равен 0, то переход к шагу 8.

Шаг 6: Присваиваем указателю на следующий элемент последнего элемента списка this адрес объекта temp. Присваиваем указателю на последний элемент this адрес объекта temp.

Шаг 7: Увеличиваем размер списка на единицу.

Шаг 8: Присваиваем указателям на первый и последний элементы списка this адрес объекта temp.

Шаг 9: Выход из функции.