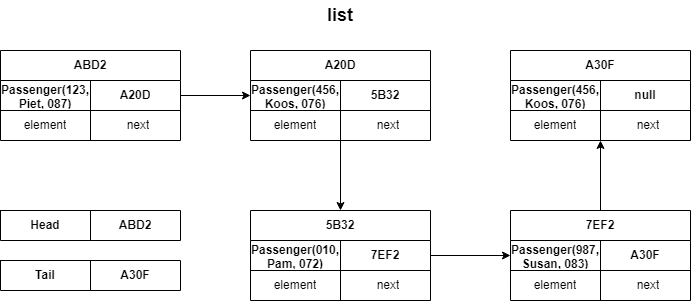
**Step 1: Draw diagram**

* 1. Write down method call in test program with variable types:

MyLinkedList listWithoutDoubles, list

listWithoutDoubles = list.checkDouble();

* 1. Draw diagram:



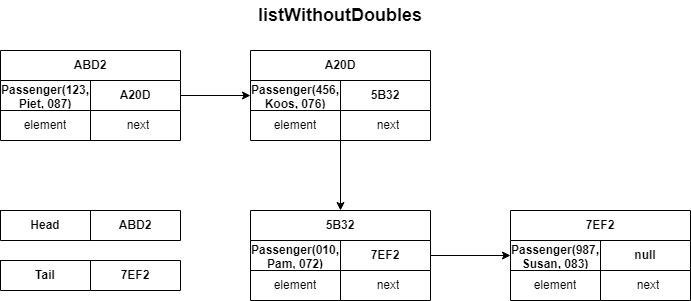
* 1. Write down Overall plan:

Create an empty list in checkDouble() called returnList.

Traverse the list and returnList with a pointer and compare the elements in list and returnList.

Only add a passenger to returnList if it has not been added already.

Return returnList that will only have one of each passenger.



**Step 2: Write down steps**

2.1 Identify data structures in terms of calling, parameter and return list

list is the calling list

listWithoutDoubles is assigned the return list in test class

returnList is the return list in MyLinkedList class

2.2 Write down call to the method and the heading of the method

Call: listWithoutDoubles = list.checkDouble(); //in test class

Heading: MyLinkedList listWithoutDoubles = new MyLinkedList();

Namne of the return variable: returnList

2.3 Write down programming steps:

Create a return list.

Create two pointers, one for each list.

Use while loop, loop through list until the end

Use while loop to loop through returnList

If !ptrList.element == ptrReturn.element

returnList.append(ptrList.element)

**Step 3: Write down the special cases:**

3.1 Make a list of everything that can go wrong:

* Calling list can be empty

3.2 Propose a solution

* Return the empty returnList anyway