

## Feuille 7 - Généricité

### 1 Trouver les erreurs

Trouver les erreurs de compilation du programme suivant. Proposer des corrections.

```

1  class A<T> {
2      T t;
3      public void setT(T t){ this.t = t; }
4      public T getT(){ return t; }
5  }
6  class B extends A<T> { int x; }
7  class C<T,T> extends A<T> { int y; }
8  class D<T,K> extends A<T> {
9      K k;
10     public void setK(K k){ this.k = k; }
11     public K getK(K k){ return k; }
12 }
13 class E { int x; }
14 class F<T> extends E { T t; }
15
16 class Test{
17     public static void main(String [] args){
18         A<Integer> aa = new A<>();
19         A<Integer> ad = new D<>();
20         E ef = new F<>();
21         aa.setT(5);
22         aa.setT("abc");
23         ((A<String>)aa).setT("abc");
24         ((A<Object>)aa).setT("abc");
25
26         ad.setK("hello");
27         ((D<Integer ,String>)ad).setK("hello");
28         ((D<Integer ,Integer>)ad).setK("hello");
29     }
30 }
```

## 2 Listes Chaînées

Considérez le programme (partiel) suivant :

```

1  class myPair<S,T> {
2      S item1;
3      T item2;
4      public myPair(S s, T t){ item1 = s; item2 = t; }
5      myPair<T,S> flip () { return new myPair<T,S>(item2 ,item1); }
6  }
7
8  class IntSomethingPair<T> extends myPair<Integer ,T> {}
9
10 class ListOfInts extends IntSomethingPair<ListOfInts> {}
11 class ListOfDoubles extends myPair<Double ,ListOfDoubles> {}
12 class ListOfT<T> extends myPair<T,ListOfT<T>> {
13     T get(int index){ /*Implement recursively*/ }
14     T get2(int index){/*Implement with loop*/}
15     void add(T t){ /*Implement with loop*/ }
16 }
17
18 class Test{
19     public static void main(String [] args){
20         ListOfInts l = new ListOfInts ();
21         l.item1 = 5;
22         l.item2.item1 = 7;
23         //Implement methods before executing the rest...
24         ListOfT<Integer> l2 = new ListOfT<>();
25         for(int i=0; i<5; i++){
26             l2.add(i);
27         }
28         for(int i=0; i<5; i++){
29             System.out.println(l2.get2(i));
30         }
31     }
32 }

```

- Trouver et corriger les erreurs de compilation.

Le but de ce programme est de donner une manière alternative d'implémenter une liste chaînée. Le type `ListOfT<T>` est un type générique qui est censé représenter une structure de données ordonnée dont chaque élément contient un objet de type `T`. Achetez l'implémentation de cette structure : donnez une méthode `add` qui permet d'ajouter un élément dans une liste de type `ListOfT<T>`, et une méthode `get(int index)` qui permet d'accéder à l'élément à position `index`.