```java
import java.io.*;

class TupleJ <T> {
    public Integer size(){
        return 0;
        }
}

class CoppiaJ <T> extends TupleJ <T>{
    private T uno;
    private T due;
    public CoppiaJ(T x, T y){
        uno=x; due=y;
        }
    public T uno(){return uno;
        }
    public T due(){return due;
        }
    public Integer size(){return 2;
        }
}

class TernaJ <T> extends CoppiaJ <T>{
    private T tre;
    public TernaJ(T x, T y, T z){
        super(x,y);
        tre=z;
        }
    public T tre(){return tre;
        }
    public Integer size(){return 3;
        }
}

 interface fun2T <T>{
    public T apply(T x, T y);
}

public class main {
    static <T> T fun_COfT (fun2T<T> f, TupleJ<T> r) throws
        IllegalArgumentException {
        int n = r.size();
        if (n==2) {CoppiaJ<T> temp = (CoppiaJ<T>) r;
                    return f.apply(temp.uno(),temp.due());
                    }
        if (n==3) {TernaJ<T> temp = (TernaJ<T>) r;
                    return f.apply(f.apply(temp.uno(),temp.due()),temp.tre
                        ());
                    };
        throw new IllegalArgumentException("main:fun_COfT-second argument
            has 0 components");
        };
    public static void main(String args[]) {
            fun2T<Integer> somma = new fun2T<Integer> (){
                public Integer apply(Integer x, Integer y){return x+y;}
                };
            fun2T<Integer> minimo = new fun2T<Integer> (){
```

```java
            public Integer apply(Integer x, Integer y){if (x<y)
                return x;
                                                        return y;}
            };
        CoppiaJ<Integer> c = new CoppiaJ<Integer>(5,10);
        TernaJ<Integer> t = new TernaJ<Integer>(5,10,15);
        try{
        int r1 = fun_COfT(somma,c);
        int r2 = fun_COfT(minimo,t);
        System.out.println(r1+"  "+r2);
            }
        catch (IllegalArgumentException e){};
        }
    }
```