Problem 1:

Java code:

```
public int gcd(int n, int m) {
            int smaller=m,d=1;
            if(n<m) smaller=n;
            for(int i=1;i<=smaller/2;i++) {
                  if(n%i==0 && m%i==0) {
                        d=i;
                  }
            }
            return d;
      }
```

Problem 2

```
package new1;
import java.util.ArrayList;
import java.util.List;
public class Main
{
    public static void main(String[] args)
    {
        int arr[] = {1,10,9,7,6,20,62,3};
        int k=19;
        sumSubSet(arr,k);
    }
    static void sumSubSet(int arr[],int k)
    {
        int n = arr.length;
        List<Integer>num = new ArrayList<>();
        for (int i = 0; i < (Math.pow(2,n)); i++)
        {
            int sum=0;
            for (int j = 0; j < n; j++)
                if ((i & ((int)Math.pow(2,j))) > 0)
                    num.add(arr[j]);
            for(int a:num)
                sum+=a;
            if (sum==k)
            System.out.println(num);
            num.clear();
        }    }
    }
```

Problem 3:

The algorithm will not work for all values, for example consider S={2,3,6,7} and k=10. Even though we have a working subset T={3,7}, the algorithm will return T={2,3}.

Problem 4:

Since we are assuming that s_{n-1} is part of T, and we are deducting/removing it from k, S' =S-{ s_{n-1} } and T'=T- s_{n-1}, therefore its true that T' will always be a solution to S'.

Examples:

S={7,2,6,3} k=10, T={7,3} then s_{n-1}=3, S'={7,2,6} k'=7, T'=7 so T' is a solution for S'.

S={8,2,4,1} k=11, T={8,2,1} then s_{n-1}=1, S'={8,2} k'=10, T'={8,2} so T' is a solution for S'.