1. Write a program that takes two string and tests whether they are anagrams of each other and prints "Anagram" if they are otherwise prints "Not Anagram".
An **anagram** is the result of rearranging the letters of a word or phrase to produce a new word or phrase, using all the original letters exactly once; for example, the word anagram can be rearranged into "nagaram"

| Sample Input | Sample Output |
|---|---|
| banana<br>abanan | Anagram |

2. Write a program that tests whether a string is palindrome or not ignoring the white spaces (i.e.tab and space) and prints "Palindrome" if it is.

| Sample Input | Sample Output |
|---|---|
| Madam Im Adam | Palindrome |
| Madam Im        Adam | Palindrome |

3. Write down a function `int find(char s[], char sub[])` which will find and return the position of the first occurrence of the string sub inside string `s`. The function returns -1 if sub doesn't occur in `s` at all. Write down a program that will read two strings `s1` and `s2` and print all occurrences of string `s2` in `s1`.  Print not found if `s2` does not occur at all in s1.

| Sample Input | Sample Output |
|---|---|
| Hello World<br>orl | 8 |