Write a function that reverses a string. The input string is given as an array of characters s.

In [12]:

```
def reverse(s):
    n=len(s)
    j=n-1
    i=0
    while(i<=j):
        temp=s[i]
        s[i]=s[j]
        s[j]=temp
        i+=1
        j-=1
    print("Output:",s)


s = ["h","e","l","l","o"]
print("Input:",s);
reverse(s)


"""Expected Time Complexity : O(n)
Expected Space Complexity : O(1)"""

```

```
Input: ['h', 'e', 'l', 'l', 'o']
Output: ['o', 'l', 'l', 'e', 'h']
```

Out[12]:

```
'Expected Time Complexity : O(n) \nExpected Space Complexity : O(1)'
```

Return the index of the first occurrence of needle in haystack, or -1 if needle is not part of haystack.

In [39]:

```
def occurrence(haystack,needle):
    for i in range(len(needle)-1):
        for j in range(len(haystack)):
            if(needle[i]==haystack[j] and needle[i+1]==haystack[j+1]):
                flag=1;

                break;
    return flag

haystack = "hello"
needle = "ll"
flag=0
flag=occurrence(haystack,needle)

if(flag==1):
    print("Output",j)
else:print("output",-1);

'''Expected Time Complexity : O(n2)
Expected Space Complexity : O(1)
'''

```

```
Output 2
```

Out[39]:

```
'Expected Time Complexity : O(n2) \nExpected Space Complexity : O(1) \n'
```

Given an array of string words. Return all strings in words which are substrings of another word in any order.

In [4]:

```
def substring(string,substr):
    sublen=len(substr)
    for i in range(len(string)-sublen):
        k=i
        for j in range(sublen):
            if(string[k]!=substr[j]):
                break;
            k+=1
        if(j==sublen-1):
            return i;
    return -1


words = ["mass","as","hero","superhero"]
for i in range(len(words)):
    for j in range(i,len(words)):
        print(substring(string[i],substr[j]))
```

```
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
```

Given a string s and an integer k, reverse the first k characters for every 2k characters counting from the start of the string.

In [56]:

```
def reverse(s,k):
    i=0
    n=len(s)
    while(i<k and k<n):
        temp=s[i]
        s[i]=s[i+1]
        s[i+1]=temp
        k=k+2
        i=k
        k=k+2
    return s

s ="abcdefg"
print("Input:",s)
reverseStr=""
s=list(s)
k = 2
reverseCh=reverse(s,k)
for i in range(len(reversech)):
    reverseStr=reverseStr+reverseCh[i]
print("Output:",reverseStr)

```

```
Input abcdefg
Output bacdfeg
```

In [ ]:

```

```