Islamic University of Technology

CSE 4810

Algorithm Engineering Lab

Lab 5

Tasnimul Hasnat

190041113

CSE 1A

April 25, 2024

Task 1

```
def expand(s,left,right):
    while left >=0 and right <len(s) and s[left] == s[right]:
        left-=1
        right+=1
    return left+1,right-1

def lps(s):
    if not s:
        return ""
    start,end=0,0
    for i in range(len(s)):
        # single center
        l1,r1 = expand(s,i,i)
        # two char center
        l2,r2= expand(s,i,i+1)

        if r1-l1 > end-start :
            start,end =l1,r1
        if r2-l2 > end-start :
            start,end =l2,r2

    return s[start:end+1]


for i in range(5):
    s = input()
    print(lps(s))
```

Here using the `lpa()` function we find the longest palindromic sub-sequence in a given string. We do this using a `expand` from the center method.

Task 2

```
def max_container(h):
    l,r= 0,len(h) -1
    max_water=0

    while l<r:
        water= min(h[l],h[r])*(r-l)
        max_water=max(water,max_water)

        if h[l]<h[r]:
            l+=1
        else:
            r-=1

    return max_water


h= [1,8,6,2,5,4,8,3,7]
print(max_container(h))

h= [1,1]
print(max_container(h))
```