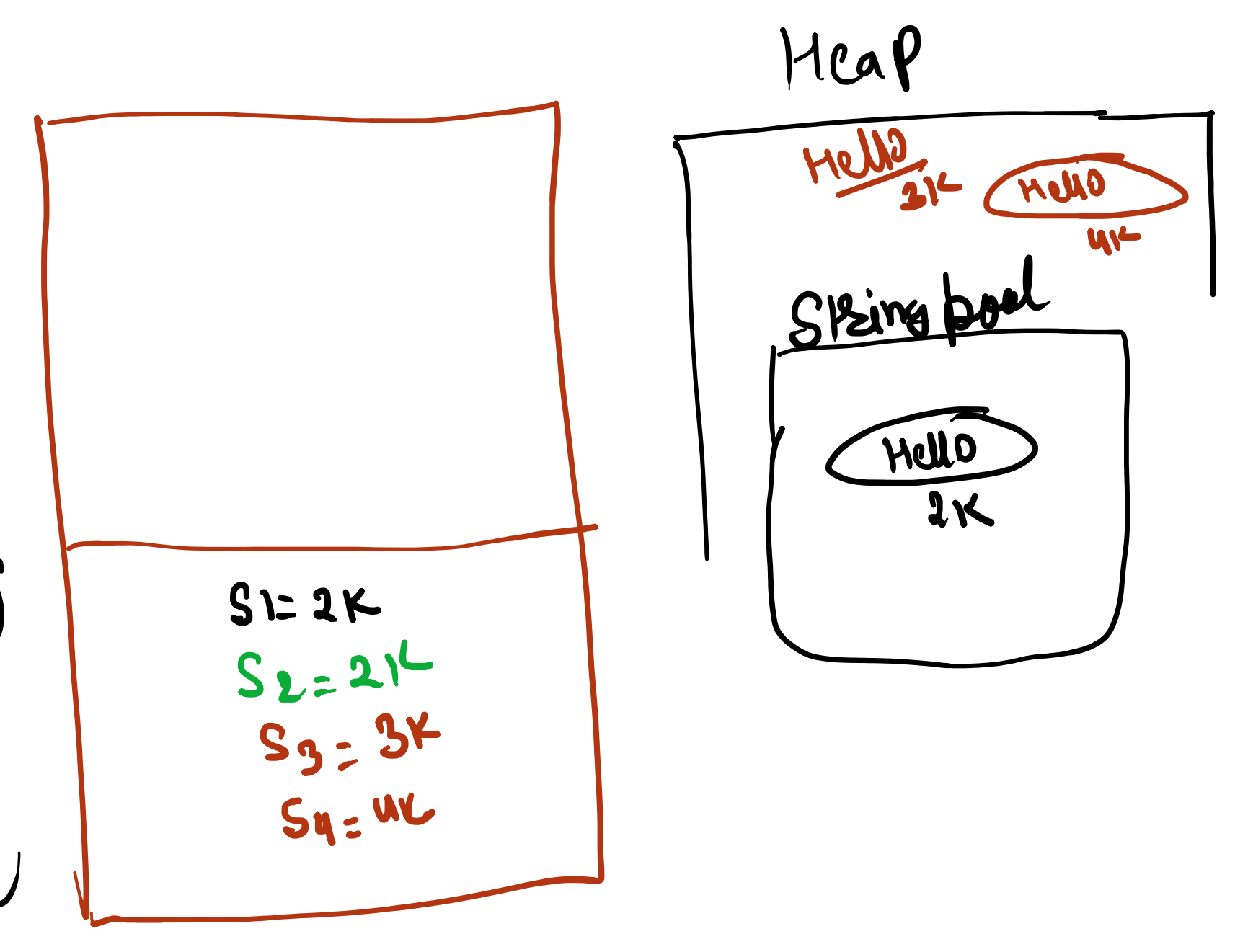
Lec-14

23/09/24 7:08 PM

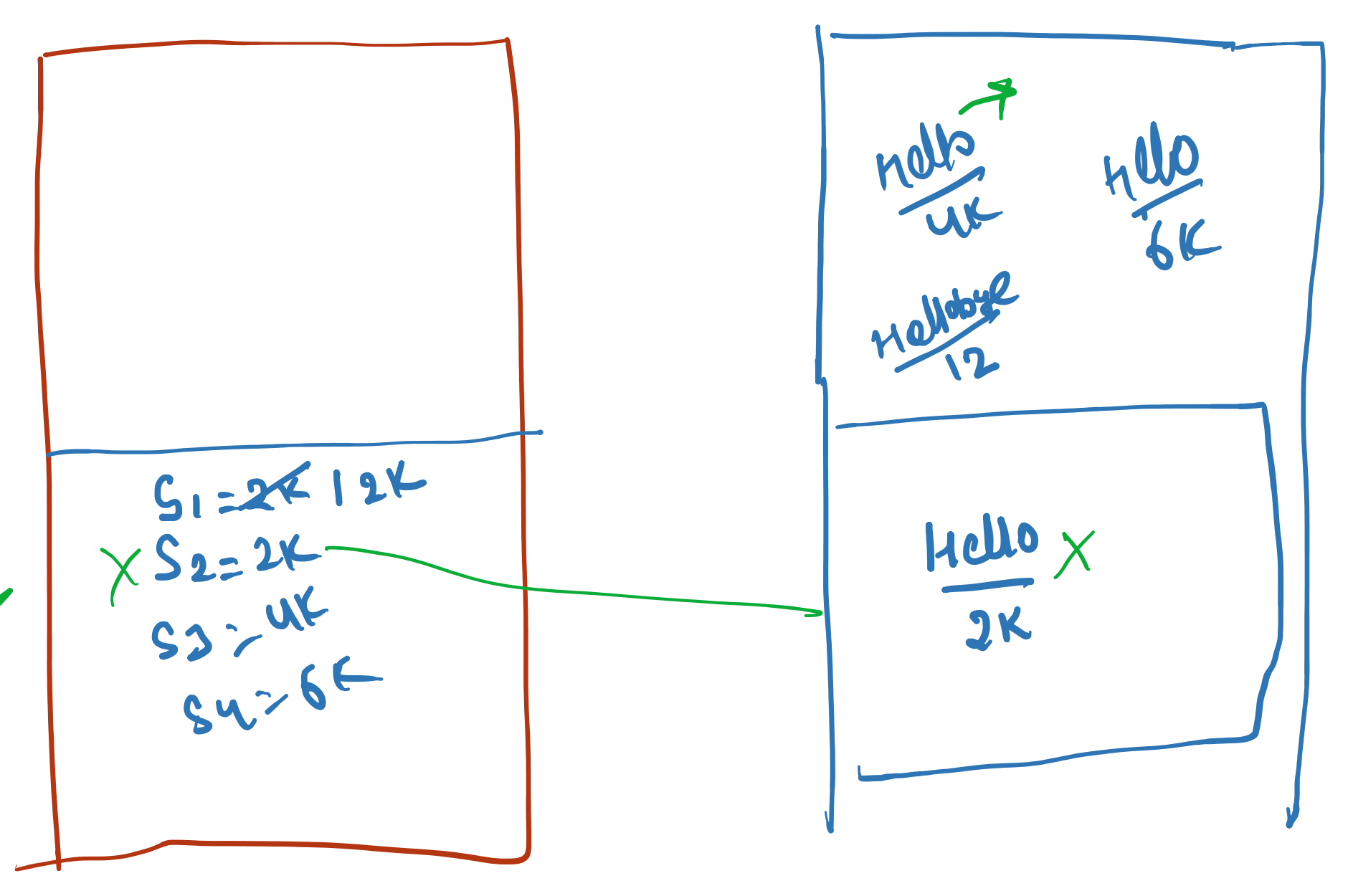
String (core)
→ Java class
→ non-primitive
→ Heap memory

```java
public static void main(String[] args) {
    // TODO Auto-generated method stub
    String s1="Hello";
    String s2="Hello";
    String s3=new String("Hello");
    String s4=new String("Hello");

}
```




```java
public static void main(String[] args) {
    // TODO Auto-generated method stub
    String s1 = "Hello";
    String s2 = "Hello";
    String s3 = new String("Hello");
    String s4 = new String("Hello");
    System.out.println(s1);
    System.out.println(s2);
    System.out.println(s3);
    System.out.println(s1 == s2);
    System.out.println(s1 == s3);
    System.out.println(s3 == s4);
    s1 = s1 + "Bye";
    System.out.println(s1);

}
```

String immutable



Palindrome

naman

nitin

Aman

namA

Lexico-graphically

Kunal Komal

Komal < Kunal

Komal
Kunal

5 ← Ankit
6 ← Ankita

Ram
Raj
Rajesh

```java
public static int compareTo(String s1, String s2) {
    int n = Math.min(s1.length(), s2.length());
    for (int i = 0; i < n; i++) {
        if (s1.charAt(i) != s2.charAt(i)) {
            return s1.charAt(i) - s2.charAt(i);
        }
    }
    return s1.length() - s2.length();
}
```

5 - 6 = -1

ankit ankita

kunal komal

abcd

a, ab, abc, abcd

b, bc, bcd

c, cd

d

hello
0 1 2 3 4

[0-1] h
[0-2] he
[0-3] hel
[0-4] hell
[0-5] hello

(1-2) e
(1-3) el
(1-4) ell
(1-5) ello

2-3 l
2-4 ll
2-5 llo

3-4 l
3-5 lo

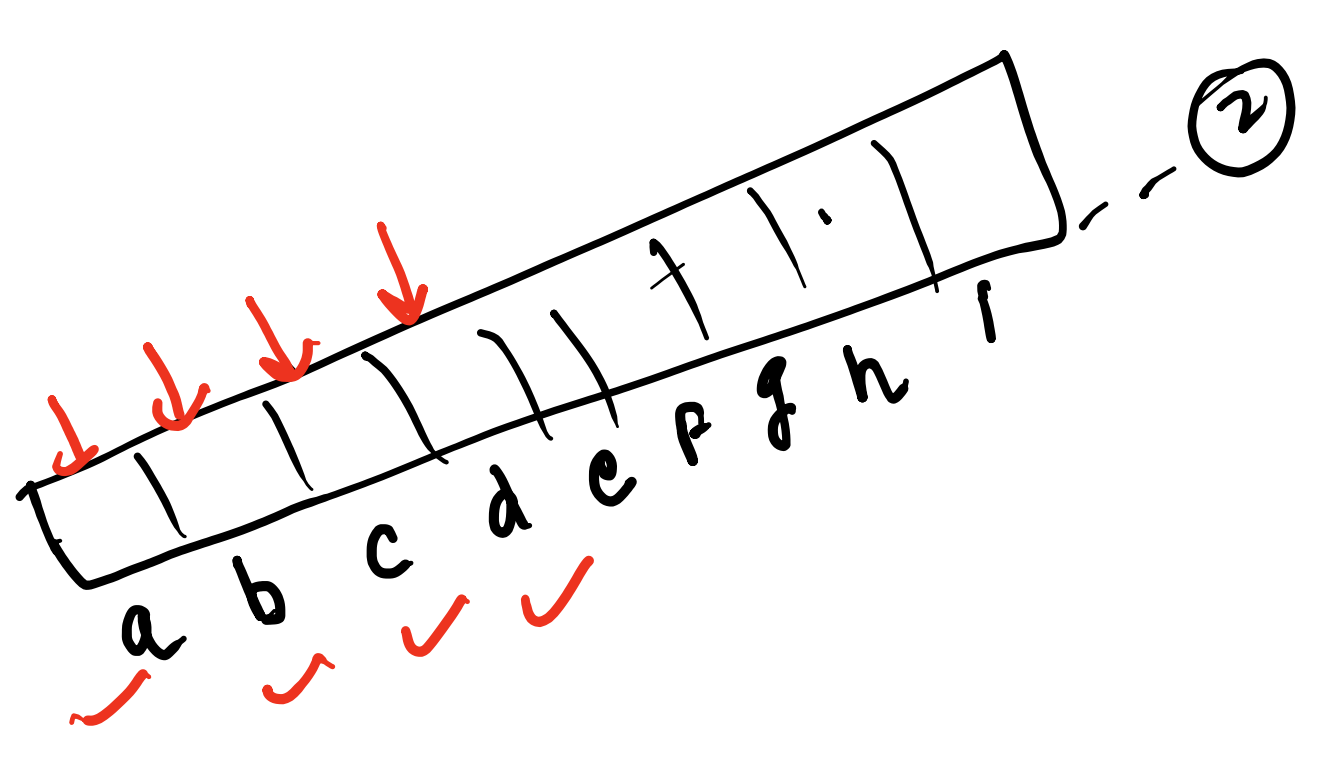
4-5 o

for(i=0; i<s.length(); i++){
for(j=i+1; j<=s.length(); j++)

abcda

'a' → 0
'b' → 1
'c' → 2
'd' → 3
'a' → 0

ch-97=
'a'-97=0
'b'-97=1 → 98-97
'c'-97=2 → 99-97



abcad

```java
public static boolean Unique(String s) {
    int[] freq = new int[26];
    for (int i = 0; i < s.length(); i++) {
        char ch = s.charAt(i);
        freq[ch - 97] = freq[ch - 97] + 1;
    }
}
```

the | sky | is | blue

blue is sky the

the sky is blue

A Good String is a string which contains only vowels (a,e,i,o,u) . Given a string S, print a single positive integer N where N is the length of the longest substring of S that is also a Good String.

Note: The time limit for this problem is 1 second, so you need to be clever in how you compute the substrings.

10^6 to 10^8

10^5 $(10^5)^2$ → 10^{10}

ans=3

count=0

int ans=0   int c=0

for(i=0; i<s.length(); i++){
    char ch = s.charAt(i)
    if(isVowel(ch)) {
        c++
    }
    else{
        ans=Max(ans,c)
        c=0
    }
}
ans = max(ans, c)