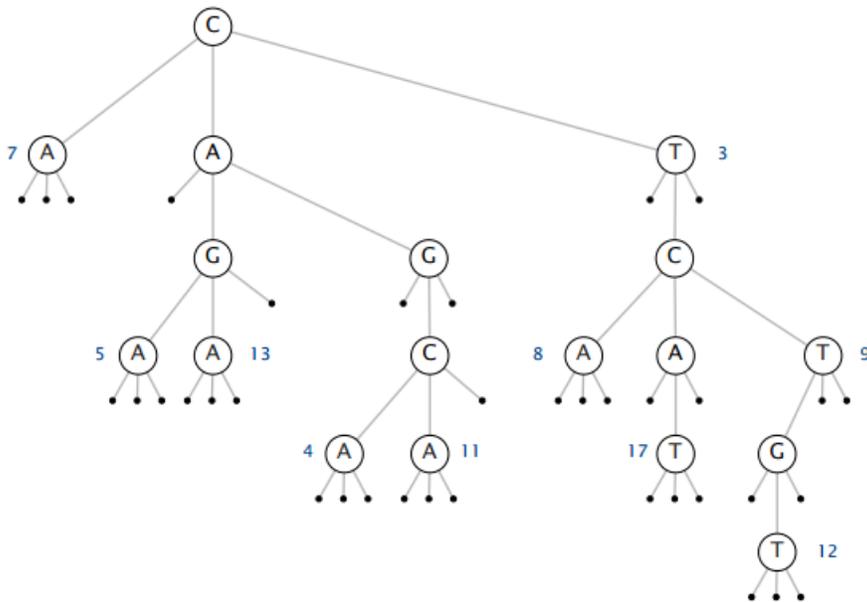# Week 10 Activity

1. **Ternary Search Trees (TST)**
   Consider the following TST, where the values are shown next to the nodes of the corresponding string keys.

   (a) list the items (in alphabetical order) that were inserted into the TST.

   (b) Insert the two strings **CGTT and TGA** into the TST with the associated values 0 and 99, respectively; update the figure above to reflect the changes.

   (c) Under what circumstances would you use a R-way Trie instead of a TST? Discuss pros and cons of each approach.

2. Burrows-Wheeler transform

(a) What is the Burrows-Wheeler transform of b a n a n a?

```
suffix[0] = b a n a n a
suffix[1] =
suffix[2] =
suffix[3] =
suffix[4] =
suffix[5] =

Sorted Suffixes
suffix[0] =
suffix[1] =
suffix[2] =
suffix[3] =
suffix[4] =
suffix[5] =
```

Write your answer in the box.

(b) Apply the Burrows-Wheeler inverse transform to find the original string

```
6
t[] = helweer
```

(c) Construct the next array as shown in the Burrows-Wheeler assignment and find the original string.

i	sorted suffixes	t	next
0			
1			
2			
3			
4			
5			
6			

Write your answer in the box.

3. **LZW Compression (Bonus Problem)**

Assume that we are working with the ASCII alphabet where a=61, b=62 in hexadecimal. The end of file character is 80. The next available code is 81.

(a) Encode the message **abbbabba** using LZW compression. Fill in the following table using new codes discovered

symbol	code
ab	81
bb	82
bba	83
abb	84
ba	85

Write the encoded message in the box.

61 62 82 81 62 61 80

(b) You receive the following LZW encoded message. Note that decoding this message involves the tricky case, where you see a code, before it is in the table. Decode the message and write the answer below.

61 62 81 83 62 80

abababab