

## Heap – Spot Question

### Problem:

A function 'concat' is implemented in such a way that it takes two strings as input and concatenates the strings in  $c * (s1+s2)$  time, where  $s1$  and  $s2$  are the lengths of the two strings and  $c$  is a constant. Given 'n' strings of varying sizes, implement a function 'minTime' to determine the least time taken for concatenating the n strings using the 'concat' function mentioned above, disregarding the order in which the strings are input. [Note: the constant factor can be ignored for calculating the time]

Function Declaration: `int minTime(char *strings[]);`

### Sample:

Input: "from", "apples", "an", "are"

Output: 29

Explanation:

First, "an" and "are" are concatenated in 5 units of time resulting in "anare", then "from" and "anare" are concatenated in 9 units of time resulting in "fromanare", then "apples" and "fromanare" are concatenated in 15 units of time resulting in "applesfromanare"

Total time =  $5 + 9 + 15 = 29$  time units, which would be the least time.