CS 463: Homework 2

Kyle Guarco

February 11, 2023

1 Solution

Analyze the runtime of the following:

```java
public static int search(String text, String pattern) {
  int N = text.length();
  int P = pattern.length();
  for (int i = 0; i < N - P + 1; i++) {
    int j;
    for (j = 0; j < P; j++) {
      if (text.charAt(i + j) != pattern.charAt(j)) {
        break;
      }
    }
    if (j == P) {
      return i;
    }
  }
  return N;
}
```

Recall that a repeated sum of ones equals its bound: $\sum_{i=1}^{n}(1) = n$
A repeated sum of some constant C: $\sum_{i=1}^{n}(C) = Cn$

We write the loops in terms of summations (the 1 indicates a constant runtime):

$$\sum_{i=0}^{N-P}\left(\sum_{j=0}^{P-1} 1\right) = \sum_{i=0}^{N-P} P \qquad \text{Reduce inner loop using `sum of ones' formula}$$

$$\sum_{i=0}^{N-P} P = P(N-P+1) \qquad \text{Repeated sum of constant } P$$

$$= -P^2 + NP + P \qquad \text{A quadratic runtime!}$$

Therefore the runtime of `search(String, String)` is $\mathbf{O}(n^2)$. Interestingly, this algorithm's runtime is dependant on the size of P (the pattern search size), rather than the input string size N.

2 Code

```java
public class Main {

  public static int search(String text, String pattern) {
    int lengthOfText = text.length();
    int lengthOfPattern = pattern.length();
    // iterate through the whole text character by character
    for (int i = 0; i < lengthOfText - lengthOfPattern+1; i++) {
      int j;
      // On every iteration we check whether the two characters are matching or not
      for (j = 0; j < lengthOfPattern; j++) {
        if (text.charAt(i + j) != pattern.charAt(j)) {
          break; // if mismatch: we break out
        }
      }
      // We have found the pattern in the text because no mismatching character has been found
      if (j == lengthOfPattern) {
        // i is the first character in the match in this case it is the index of the character T
        return i;
      }
    }
    return lengthOfText; //The given text does not contain the given pattern
  }

  public static void main(String[] args) {
    String text = "THIS IS A TEST";
    String pattern = "TEST";

    int result = search(text, pattern);

    System.out.println("result: " + result);
  }
}
```