sheet 1

Yaseen Islam Mohamed
9032

1 a palindrome(word, start, end)
    if end lessThan or equal start
        return "True"
    else if word[start] not equal word[end]
        return "False"
    else
        return palindrome(word, start+1, end-1)

b Binary Search (Key, Arr, start, end)
    mid = (start+end) / 2
    if Array[mid] is equal to Key
        return 1
    else if end lessThan start
        return 0
    else if Key greaterThan Arr[mid]
    return BinarySearch (Key, Arr, mid, end)
    else if Key lessThan Arr[mid]
        return BinarySearch [Key, Arr, start, mid)

③ ⓐ for(i=0; i<=n*n; i+=n/2) { O(2n)

print("a");

for(j=n; j>=0; j-=2) {

print("b");

}

}

$i = 0, n/2, 2\times n/2, 3\times n/2, 4\, n/2, ---, k_1\, n/2$

Base case

$k_1\, n/2 = n^2$

$\boxed{k_1 = 2n}$

$j = n, n-2, n-4, n-2k_2$

$n - 2k_2 = 0$

$2k_2 = n$

$k_2 = n/2$

$T(n) = O(n^2)$

(b)

```
for(i=0; i<=n/2; i++){      O(n/2)
    print("a");
    for(j=1; j<=n; j*=5){
        print("b");
    }
    for(k=1; k<=n; k+n-10){
        print("c");
    }
}
```

$i = 0, 1, 2, 3, \ldots, n/2$

$j = 1, 5, 25, 125, \ldots, (5)^Z$

$(5)^Z = n$

$Z = \log_5 n$

$k = 1, 1 + (n-10), 2(n-10)+1, 3(n-10)+1, \ldots, Z(n-10)+1$

$Z(n-10)+1 = n$

$Z = \frac{n-1}{n-10} \simeq Z = n/n = 1$

$T(n) = n/2 \cdot (\log_5 n + 1)$

$T(n) = n\lg n + n = O(n\lg n)$

5

$T(n)$ $\quad n$

$T(n/2) \quad T(n/4) \quad T(n/8)$ $\quad \frac{7n}{8}$

$T(n/4) \quad T(n/8) \quad T(n/16) \quad T(n/8) \quad T(n/16) \quad T(n/32) \quad T(n/16) \quad T(n/32) \quad T\left(\frac{n}{64}\right)$

$\frac{49n}{64}$

$T\left(\frac{n}{2^k}\right)$

$\frac{n}{2^k} = 1$

$k = \lg n$

$$T(n) = n\sum_{k=0}^{\lg n}\left(\frac{7}{8}\right)^k$$

$$= n\left(\frac{1-\left(\frac{7}{8}\right)^{\lg n}}{1-7/8}\right)$$

$7 = 2^{\lg 7} = 2^{2.81}$

$$= n\left(\frac{1-\frac{2^{\lg n^{2.81}}}{2^{\lg n^3}}}{1-7/8}\right)$$

$$= n\left(\frac{1-\frac{1}{n^{0.19}}}{1/8}\right)$$

$$= 8n - 8n^{0.81}$$

$$\simeq n$$

$\boxed{T(n) = O(n)}$

6 (a) $3T(n/2) + n\lg n$

$a = 3 \quad b = 2 \quad f(n) = n\lg n$

$\Theta(n^{\log_b a}) = \Theta(n^{1.5})$

$n^{1.5} > n\log_2 n$

Then $T(n) = \Theta(n^{1.5})$

(b) $4T(n/2) + n^2\sqrt{n}$

$a = 4, b = 2 \quad f(n) = n^2\sqrt{n}$

$n^{\log_2 4}$

$n^2 < n^2\sqrt{n}$

$af(n/b) \leq cf(n)$

$4\left(\frac{n}{2}\right)^2\sqrt{n/2} = c\,n^2\sqrt{n}$

$c = \frac{1}{\sqrt{2}} \quad \text{if } c < 1$

Then $T(n) = \Theta(n^2\sqrt{n})$

[4] 8^{8^n}

$\log_8 \sqrt{n} < \sqrt{\lg n} < 2^{\lg n} < \lg n! < \lg n^n = n\lg n < n^2 3^n$

$< 8^n < 16^n < n! < n^n < 8^{8^n}$

[2]

$cn^2 < n\lg n$

$cn < \log_2 n$

$c < \frac{\log_2 n}{n}$

$\lim_{n \to \infty} \left(\frac{\log_2 n}{n}\right) = \lim \left(\frac{\frac{1}{n} \cdot \frac{1}{\ln(2)}}{1}\right)$

$\boxed{c = \text{Zero}}$

There is no c such That nLogn would Be greater Than n^2

Then using $\Theta(n\log n)$ is better for Time complexity But n^2 will b more efficient for small size input

example $10^8 n^2$ $10^3 n\log n$

$n = 4$

$10^8 \times 16 = 16$ $10^3 \times 4 \times 2 = 8$