CS170–Spring 2020 — Homework 1 Solutions

Blackteaxx

4. Asymptotic Complexity Comparsions

(a)

$$12 < \log_2 n < n^{\frac{1}{3}} < \sqrt{n} < 2^{\log_2 n} < n^3 < 2^{\sqrt{n}} < 2^n < 3^n$$

(b) (a) $f = \Theta(g)$

(b) $f = O(g)$

(c) $f = \Omega(g)$

(d) $f = \Theta(g)$

5. Computing Factorials

(a)

$$f(N) = N \log N$$

because the following inequality:

$$\prod_{i=\frac{N}{2}}^{N} i \leq N! \leq N^N \Rightarrow \frac{N}{2}^{\frac{N}{2}} \leq N! \leq N^N$$

so, the digits are like the following:

$$\frac{N}{2} \log \frac{N}{2} \leq \log N! \leq N \log N$$

We can throw the constant items, the get the Θ asymptotic complexity

(b) Algorithm Description is following:

Algorithm 1 Naive Factorials Algorithm

Input: n
Output: $n!$
  if $n \leq 1$ **then**
    return 1
  else if $n > 1$ **then**
    return $Fact(n-1) * n$
  end if

and the runtime analysis is following:

$$T(N) = \log N \times (N-1)\log(N-1) + T(N-1)$$

$$T(N) = \sum_{i=1}^{N} i^2 \log i$$

6.

(a) The naive polynomial algorithm work through every orders and compute the precise values:

Algorithm 2 Naive Polynomial Algothrim

Input: $[a_0, \ldots, a_n], x$
Output: $p(x)$
  $res \leftarrow 0$
  for each $i \in [0, n]$ **do**
    $res \leftarrow res + a_i \times x^i$
  end for
  return res

---

and the runtime analysis is following:

$$T(n) = \sum_{i=0}^{n}(i+2) = \mathcal{O}(n^2)$$

(b) we can evaluate the polynomial value as the expression goes:

$$p(x) = a_0 + x(a_1 + x(a_2 + \cdots + x(a_{n-1} + xa_n) \ldots))$$

Owing to this expression, we can maintain a variable and calculate value from the inner to outer place.

Algorithm 3 Improved Polynomial Algorithm

Input: $[a_0, \ldots, a_n], x$
Output: $p(x)$
  $res \leftarrow a_n$
  for each $i \in [n-1, \ldots 0]$ **do**
    $res \leftarrow a_i + res \times x$
  end for
  return res

---

and the runtime analysis is following:

$$T(n) = 2 \times n = \mathcal{O}(n)$$