2 Lecture - CS502

Important Mcqs

- 1. What is the purpose of using asymptotic notation in the analysis of algorithms?
 - a) To focus on the growth rate of the function
 - b) To consider constant factors and lower order terms
 - c) To provide precise measurements of algorithm efficiency

d) To identify the fastest algorithm

Answer: a) To focus on the growth rate of the function

Which asymptotic notation represents the upper bound of a function?

- a) Big O
- b) Omega
- c) Theta
- d) None of the above

Answer: a) Big O

Which asymptotic notation represents the lower bound of a function?

- a) Big O
- b) Omega
- c) Theta
- d) None of the above

<mark>Answer: b) Omega</mark>

Which asymptotic notation represents both the upper and lower bounds of a function?

- a) Big O
- b) Omega
- c) Theta
- d) None of the above
- Answer: c) Theta

Which of the following statements is true about Big O notation?

- a) It represents the exact running time of an algorithm
- b) It represents the best-case running time of an algorithm
- c) It represents the worst-case running time of an algorithm
- d) It represents the average-case running time of an algorithm

Answer: c) It represents the worst-case running time of an algorithm

Which of the following notations is used to describe the best-case running time of an algorithm?

- a) Big O
- b) Omega
- c) Theta

d) None of the above

Answer: b) Omega

Which of the following notations is used to describe the average-case running time of an

algorithm? a) Big O b) Omega c) Theta d) None of the above Answer: d) None of the above (average-case running time is typically not described using asymptotic notation)

Which of the following functions has a higher growth rate: $f(n) = n^2$ or $g(n) = 2^n$?

a) $f(n) = n^2$ b) $g(n) = 2^n$ c) They have the same growth rate Answer: b) $g(n) = 2^n$

Which of the following functions has a lower growth rate: $h(n) = \log n$ or j(n) = n?

a) h(n) = log n
b) j(n) = n
c) They have the same growth rate
Answer: a) h(n) = log n

Which of the following notations can be used to describe an algorithm with a constant running time?

a) Big O

b) Omega

c) Theta

d) None of the above

Answer: c) Theta (since constant time is both an upper and lower bound)