Ambedkar College of Arts and Science, Wandoor — Class Test 2

Data Structures and Algorithms (MAT5EJ302(1)) — B.Sc. Mathematics Honours, Semester V

Name: ______________________ ID No.: ______________________ Time: 30 Mins **Max Marks: 30**

---

8 questions, 5 marks each — answer any 6 (best 6 count, ceiling 30). Show all working.

Q1. Consider the recursive algorithm

```
function fib1(n)
  if n = 0:  return 0
  if n = 1:  return 1
  return fib1(n-1) + fib1(n-2)
```

(a) Write the values of $F_0, F_1, \ldots, F_8$. [1 mark]

(b) Let $T(n)$ denote the number of basic steps performed by `fib1(n)`. Write the recurrence relation satisfied by $T(n)$ for $n > 1$. [1 mark]

(c) Using your recurrence from (b), prove that $T(n) \geq F_n$, and hence explain why the running time of `fib1` is exponential in n. [3 marks]

[5 M]

Q2. Consider the iterative algorithm

```
function fib2(n)
  if n = 0:  return 0
  create an array f[0...n]
  f[0] = 0;  f[1] = 1
  for i = 2 to n:
    f[i] = f[i-1] + f[i-2]
  return f[n]
```

(a) Explain why `fib2` avoids the repeated computations present in `fib1`. [1 mark]

(b) Determine the time complexity of `fib2` when each addition is treated as one basic operation. [1 mark]

(c) Determine the space complexity of `fib2` when the entire array $f[0 \ldots n]$ is stored. [1 mark]

(d) F_n requires $\mathcal{O}(n)$ bits to represent once n is large. Explain why this means the assumption that each addition in `fib2` takes constant time is *not* truly accurate for large n. [2 marks]

[5 M]

Q3.

(a) Simplify the following expression, giving the tightest standard asymptotic upper bound: $7n^3 + 4n^2 + 100n + 50$. [1 mark]

(b) Arrange the following functions in increasing order of growth: $n^2,\ \log n,\ 2^n,\ n \log n,\ n$. [1 mark]

(c) Using the formal definition of Big-O, show that $3n^2 + 5n + 7 = \mathcal{O}(n^2)$. Clearly specify suitable constants c and n_0. [2 marks]

(d) *True or False, with a one-line reason:* "If $T(n) = \mathcal{O}(n^2)$, then it must also be true that $T(n) = \mathcal{O}(n^3)$." [1 mark]

[5 M]

Q4.

(a) Two integers each contain n digits. In the standard grade-school addition algorithm, explain why their addition takes $\mathcal{O}(n)$ time. [1 mark]

(b) In the standard grade-school multiplication algorithm, each digit of one number is multiplied by every digit of the other number. Determine the asymptotic running time for multiplying two n-digit integers, and justify your answer. [2 marks]

(c) Which grows faster as $n \to \infty$: n^2 or $n \log n$? Justify your answer using the limit of the ratio of the two functions. [2 marks]

[5 M]

Q5.

(a) Compute 347 mod 12. [1 mark]

(b) Compute -23 mod 5. [1 mark]

(c) Compute (37×58) mod 12. You may reduce the factors modulo 12 first. [1 mark]

(d) Does 8 have a multiplicative inverse modulo 12? State the criterion you used, and justify your answer *without* computing the inverse explicitly. [2 marks]

[5 M]

Q6. Compute 7^{13} mod 20 using repeated squaring. Your solution should:

(a) Express 13 in binary. [1 mark]

(b) Compute the required powers of 7 modulo 20 using repeated squaring (i.e. $7^1, 7^2, 7^4, 7^8$ mod 20). [2 marks]

(c) Combine the required powers to obtain the final answer modulo 20. [2 marks]

[5 M]

Q7. The recursive algorithm $\texttt{ModExp}(x, y, N)$, computing x^y mod N, is defined as: if $y = 0$ return 1; else let $z = \texttt{ModExp}(x, \lfloor y/2 \rfloor, N)$, and return z^2 mod N if y is even, or xz^2 mod N if y is odd.

(a) Trace $\texttt{ModExp}(3, 13, 17)$ completely, showing the argument and return value of every recursive call. [3 marks]

(b) Let n be the number of bits needed to represent N, and suppose each modular multiplication costs $\mathcal{O}(n^2)$. Using this, explain why `ModExp` runs in $\mathcal{O}(n^3)$ time overall. [2 marks]

[5 M]

Q8. Consider the five expressions $n^{1.5}$, 2^n, $n \log n$, $\log(n^n)$, $n!$.

(a) Two of these five expressions are, in fact, exactly equal for all n. Identify which two, and show why. [2 marks]

(b) Arrange the remaining *distinct* growth rates in increasing order, justifying each comparison you make. [3 marks]

[5 M]

— End of Question Paper —