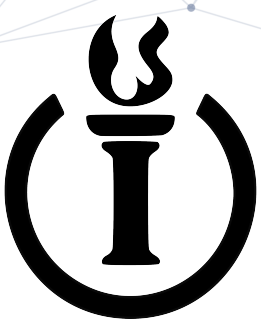

International Computer Science Competition

Final Round Solution

This document contains basic example solutions which are by no means complete or cover all possible correct solutions. The solutions serve as a basic pointer for how a given problem can be approached. For most problems, more than one approach can lead to full points.

Q1. What is the time complexity of finding the minimum element in an unsorted array of size n?

- **A.** O(1)
- **B.** O(log n)
- **C.** O(n)
- **D.** O(n²)

Answer: C – O(n)

Q10. What is the Boolean operation that outputs 1 only when inputs are different?

Answer: XOR

Q11. Write a function factorial$(n : \text{int}) \to \text{int}$ that returns the factorial of n (n!).

Answer:

```
def factorial(n):
    if n <= 1:
        return 1
    return n * factorial(n - 1)
```

Q12. Write a function is_prime$(n : \text{int}) \to \text{bool}$ that checks if a number is prime. Return True if prime, False otherwise.

Answer:

```
def is_prime(n):
    if n < 2:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True
```

Q13. Which Boolean formula expresses $F_{2,4}(a, b, c, d)$ ("at least 2 of 4")?

- **A.** (a ∧ b) ∨ (a ∧ c) ∨ (a ∧ d) ∨ (b ∧ c) ∨ (b ∧ b) ∨ (d ∧ d)
- **B.** (a ∨ b) ∧ (c ∨ d)
- **C.** (a ∧ b) ∨ (c ∧ d) ∨ ((a ∨ b) ∧ (c ∨ d))
- **D.** (a ∨ b) ∧ (a ∨ c) ∧ (a ∨ d) ∧ (b ∨ c) ∧ (b ∨ d) ∧ (c ∨ d)

Answer: C – (a ∧ b) ∨ (c ∧ d) ∨ ((a ∨ b) ∧ (c ∨ d))

Q14. What is contextual diversity (v) in the von Mises-Fisher framework?

- **A.** Equal to κ
- **B.** Equal to $\frac{1}{k}$

C. Equal to $\frac{\kappa}{2}$

D. Equal to $log(\kappa)$ **Answer: B – Equal to $\frac{1}{k}$**

Q15. Which sorting algorithm has the best worst-case time complexity?

A. Quick Sort

B. Bubble Sort

C. Merge Sort

D. Selection Sort

Answer: C – Merge Sort

Q16. How many gates are needed at least to compute the "at least k" function $F_{2,6}$?

Answer: 9

Q17. In Huffman coding, which type of characters are assigned the shortest codes?

Answer: most frequent

Q18. Given two positive numbers in binary: A = 1011 0110 and B = 1101 0011. Calculate the sum of A and B in binary.

Answer: 11000 1001

Q19. A mystery box contains red and blue marbles in equal proportions. What is the entropy of randomly selecting one marble?

A. 0 bits

B. 0.5 bits

C. 1 bit

D. 2 bits

Answer: C – 1 bit

Q2. Which data structure uses the Last In First Out principle?

A. Queue

B. Stack

C. Tree

D. Graph

Answer: B – Stack

Q20. In greedy decoding for language models, how is the next token selected?

A. Randomly

B. The token with highest probability

C. The token with lowest perplexity

D. Using beam search

Answer: B – The token with highest probability

Q21. What is the output of NOT(NAND(1,1))?

Answer: 1

Q22. Write a function find_max(lst : list[int]) $\rightarrow$ int that finds the maximum element in a list.

Answer:

```
def find_max(lst):
    return max(lst)
```

Q23. What do Nagata et al. (2025) argue makes masked language models observe Zipf's Law with fewer parameters than autoregressive models?

A. They train faster

B. They use bidirectional context

C. They have better optimization

D. They use more memory

Answer: B – They use bidirectional context

Q24. What is the result of 5 XOR 3 in binary arithmetic?

Answer: 6

Q25. Write a function maxPaintProfit($prices$: List[int], k : int) $\rightarrow$ int that returns the maximum profit from painting at most k houses, where you cannot paint two adjacent houses. Each house i gives profit prices[i].

Answer:

```
def maxPaintProfit(prices, k):
    n = len(prices)
    if n == 0 or k == 0:
        return 0

    # dp[i][j] = max profit painting at most j houses from first i houses
    dp = [[0] * (k + 1) for _ in range(n + 1)]

    for i in range(1, n + 1):
        for j in range(1, k + 1):
            # Don't paint house i-1
            dp[i][j] = dp[i-1][j]
```

```
            # Paint house i-1 (can't paint i-2)
            if i >= 2:
                dp[i][j] = max(dp[i][j], dp[i-2][j-1] + prices[i-1])
            else:
                dp[i][j] = max(dp[i][j], prices[i-1])

    return dp[n][k]
```

Q26. What search algorithm is most efficient for finding an element in a sorted array?

- **A.** Linear Search
- **B.** Binary Search
- **C.** Depth-First Search
- **D.** Breadth-First Search

Answer: B – Binary Search

Q27. What is the result of left-shifting in binary the number 5 by 2 positions (5 « 2)? Give the result in the decimal system.

Answer: 20

Q28. Write a function count_vowels(s : str) $\rightarrow$ int that counts the number of vowels (a,e,i,o,u) in a string (case-insensitive).

Answer:

```
def count_vowels(s):
    vowels = 'aeiouAEIOU'
    return sum(1 for c in s if c in vowels)
```

Q29. In the von Mises-Fisher distribution for measuring contextual diversity, what does high concentration (high κ) indicate about word usage?

Answer: low diversity

Q3. In a neural network, what does the activation function do?

- **A.** Computes the weighted sum of inputs
- **B.** Introduces non-linearity to the model
- **C.** Initializes the weights
- **D.** Calculates the loss

Answer: B – Introduces non-linearity to the model

Q30. Write a function has_arithmetic_subseq($nums$: List[int], min_length : int) $\rightarrow$ bool that checks if a list contains any contiguous arithmetic subsequence of at least min_length elements. An arithmetic sequence has constant difference between consecutive elements.

Answer:

```
def has_arithmetic_subseq(nums, min_length):
    n = len(nums)
    if n < min_length:
        return False
    for i in range(n - min_length + 1):
        diff = nums[i + 1] - nums[i]
        is_arith = True
        for j in range(i + 1, i + min_length):
            if nums[j] - nums[j - 1] != diff:
                is_arith = False
                break
        if is_arith:
            return True
    return False
```

Q4. What is the purpose of a bias term in a neural network?

- **A.** To prevent overfitting
- **B.** To add an offset to the weighted sum
- **C.** To normalize the inputs
- **D.** To reduce training time

Answer: B – To add an offset to the weighted sum

Q5. Which gate is functionally complete on its own?

- **A.** AND
- **B.** OR
- **C.** XOR
- **D.** NAND

Answer: D – NAND

Q6. In variable-length encoding, what is the primary reason for assigning shorter codes to frequent characters?

- **A.** To make decoding faster
- **B.** To reduce average bits per character
- **C.** To prevent errors
- **D.** To simplify the encoding algorithm

Answer: B – To reduce average bits per character

Q7. What is the maximum number of nodes in a complete binary tree of height 3?

Answer: 15

Q8. In Big O notation, what is the dominant term in: $5n + 7$?

Answer: 5n

Q9. What layer type comes between the input and output layers in a multi-layer perceptron?

Answer: hidden layer