Practice Exam 2

Take this exam under exam conditions. The questions you see may be harder or easier than what you'll see on the actual exam.

1. Induction proof

    - Prove that $n! > 2^n$ for $n \geq 4$

2. Probability Question

    - Let A and B be two events. Suppose that the probability that neither A or B occurs is 2/3. What is the probability that one or both occur?

      (a) 2 / 3

      (b) 1 / 2

      [✓ (c) 1 / 3]

      (d) 1 / 4

3. Counting Question

    - There are 6 men and 7 women in a ballroom dancing class. If 4 men and 4 women are chosen and paired off, how many pairings are possible?

      (a) 4!

      (b) P(6, 4) * P(7, 4) * 4!

      [✓ (c) C(6, 4) * C(7, 4) * 4!]

      (d) C(6, 4) + C(7, 4) + 4!

4. Probability Question

    - Suppose that $P(A) = 0.4$, $P(B) = 0.3$, and $P((A \cup B)^C) = 0.42$. Are A and B independent? Why?

5. Pointer Question

- Determine the output of the following code:

  (a) 20

  (b) 25

  (c) 30

  (d) 35

C++

```
// For question 5
int f(int* n, int m) {
    *n = 10;
    m = 10;
    return *n + m;
}

int main() {
    int n = 5;
    int m = 5;
    int res = f(&n, m);
    std::cout << res + n + m << std::endl;
}
```

6. Counting Question

   - How many arrangements are there of the word PROBABILITY?

7. Algorithm Analysis

   - What is the worst-case runtime of the following algorithm?

     (a) $O(n^2)$

     (b) $O(n * m)$

     (c) $O(m^2)$

     (d) $O(n + m)$

C++

```
// For question 7
void f(int* n, int* m, int n_size, int m_size) {
    for (int i = 0; i < n_size; i++) {
        for (int j = 0; j < m_size; j++) {
            // Some O(1) operation here
        }
    }
}
```

8. Algorithm Analysis

   - What is the worst-case runtime of the following algorithm? This is not C++ syntax but the meaning should be clear.

     (a) $O(n)$

     (b) $O(n^2)$

     (c) $O(\log n)$

     (d) $O(\sqrt{n})$

C++

```
// For question 8
void f(int* m, int m_size) {
    for (int i = 0; i * i < n; i++) {
        for (int j = 0; j * j < n; j++) {
            // Some O(1) operation
        }
    }
}
```

9. Pointer Question

   - What is the output of the following code?

     (a) 28 12 3

     (b) 41 17 7

     (c) 48 24 11

(d) 40 12 7

C++

```cpp
// For question 9
int g(int* n, int m) {
    *n += 12;
    m = 6;
    return *n + 4 * m;
}          17        6

int f(int* n, int& m) {
    m += 4;
    *n = 5;
    return g(n, m);
}            5  7

int main() {
    int n = 12;
    int m = 3;
    std::cout << f(&n, m) << " " << n << m << std::endl;
}                   41              17   7
```

10. Probability Question

    - Suppose 100 people all toss a hat into a box and then proceed to randomly pick out a hat. What is the expeced number of people who get their own hat back?

11. Move Zeroes: Given an array nums, write a function to move all 0's to the end of it while maintaining the relative order of the non-zero elements. Do this in-place. Furthermore, the optimal algorithm should run in $\Theta(n)$.

Example: [0, 2, 0, 1, 0] -> [2, 1, 0, 0, 0]

```
void moveZeroes(int nums[], int numsSize) {
    // TODO
}
```

12. Recursion: Given an array nums, find the length of the longest sequence of zeroes recursively. (Hint: You are allowed to use the `std::max` function from STL.)

    Example: maxZeroLength([0, 0, 1, 0, 0, 0], 6, 0) = 3

C++

```cpp
int maxZeroLength(int nums[], int len, int startIdx) {
    // TODO
}
```