Write your name and solutions clearly on this page (and back if necessary). There are 10 points for this quiz.

Problem 1. (5 pts)
Give as good a big-O estimate as possible for the following three functions

1. (1 pts) $f(n) = (n! + 2^n)(n^3 + \log(n^2 + 1))$.
2. (2 pts) $f(n) = (2^n + n^2)$.
3. (2 pts) $f(n) = (n^{10000000} + 1.000001^n)$.

Solution: We need to determine the dominant term for each function. For (1), the dominant term is $n!n^3$, therefore, $f(n) = O(n!n^3)$.
Likewise, the big-O estimates for (2) and (3) are $O(2^n)$ and $O(1.000001^n)$, respectively.

Problem 2. (5 pts)

```
Prog(input: a(1), ..., a(n), n distinct integers)
    for i=1 to n
        M=a(i)
        for j=1 to n except i
            if a(j) > M goto next
        endfor
        return M
goto next:
```

1. (1 pt) What does this algorithm return?

2. (2 pts) What is the worst-case time complexity of this algorithm? (Briefly justify your answer.)

3. (2 pts) What is the best-case time complexity of this algorithm? (Briefly justify your answer.)

Solution: (1) Maximum number in the input sequence of numbers.
(2) $O(n^2)$ (The maximum element in the sequence is the last element in the sequence.)
(3) $O(n)$ (The maximum element is the first element in the sequence.)