Data Structures and Algorithms (CS F211) – T1

Prof.N.L.Bhanu Murthy

BITS Pilani
Hyderabad Campus

Problem-1 Use the definition of Big-Oh to prove that $0.001n^3 - 1000n^2 \log n - 100n + 5$ is $O(n^3)$.

Problem-2 Prove or disprove each of the following.

1. $f(n) = O(g(n))$ implies $g(n) = O(f(n))$.
2. $f(n) + g(n) = \Theta(min(f(n), g(n))$.
3. $f(n) = O(g(n))$ implies $g(n) = \Omega(f(n))$.

Problem-3 Rank the following functions by asymptotic growth rate in non-decreasing order:
$2^{64}-1$, n^3, $0.0001n^2$, $10000n$, $\log n^2$, $2^{\log n}$, $n \log n$, $n2^n$, 2^{1000}, n, $n^2 \log n$, 2^n, $\log n$, n^{100}, 4^n, $\log n^3$, n^n.

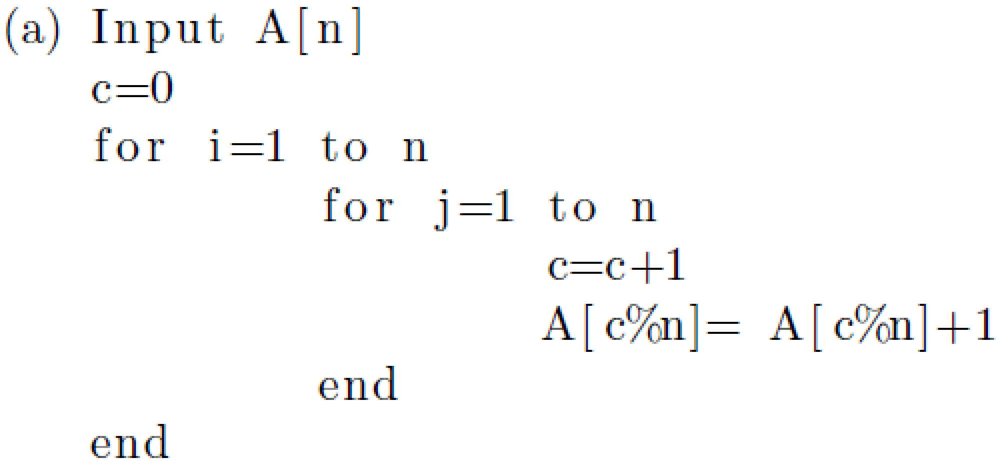
Problem-4 Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$?

Problem-5 Use the definition of Big-Oh to prove that $n^{1+0.001}$ is not $O(n)$.

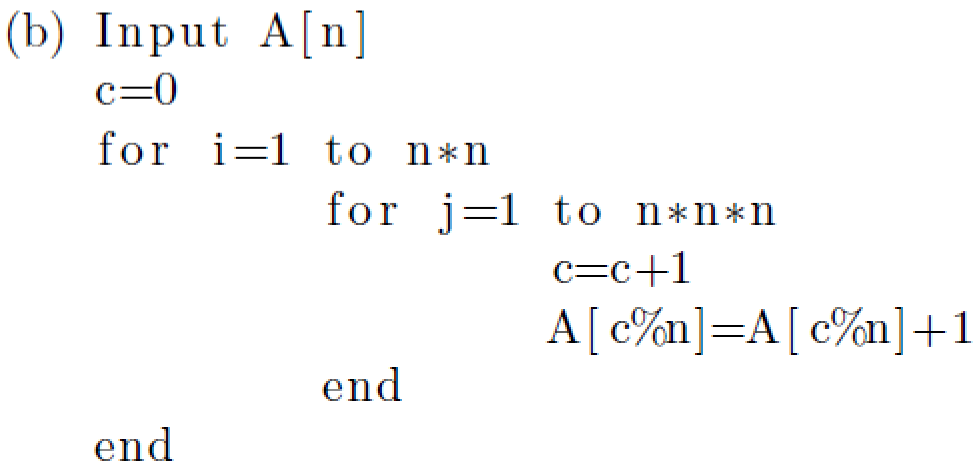
Problem-6 Express the function $n^3/1000 - 100n^2 - 100n + 3$ in terms of Θ-notation

Problem-7 Prove that $o(g(n)) \cap \omega(g(n))$ is the empty set.

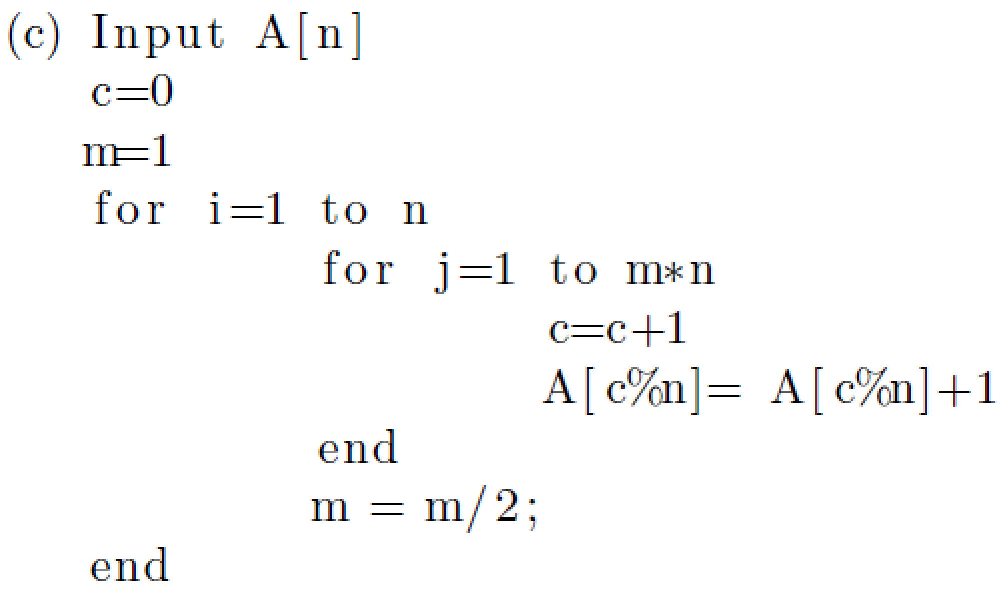
Problem-8 Let processing time of an algorithm of Big-Oh complexity $O(f(n))$ be directly proportional to $f(n)$. Let three such algorithms A, B, and C have time complexity $O(n^2)$, $O(n^{1.5})$, and $O(n \log n)$ respectively. During a test, each algorithm spends 10 seconds to process 100 data items. Derive the time each algorithm should spend to process 10,000 items.



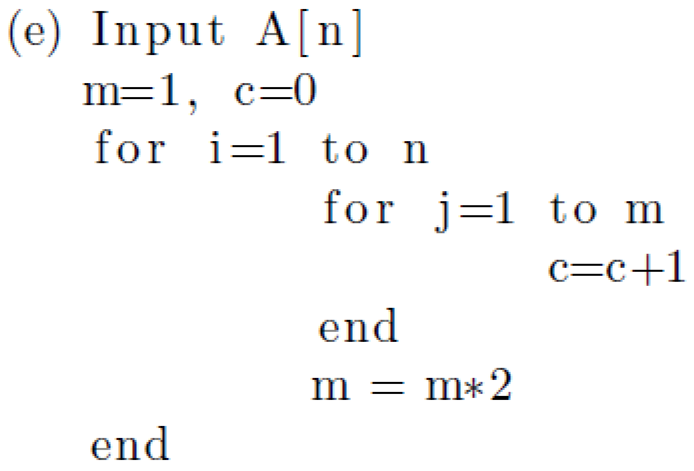
```
(a) Input A[n]
    c=0
    for i=1 to n
        for j=1 to n
            c=c+1
            A[c%n]= A[c%n]+1
        end
    end
```



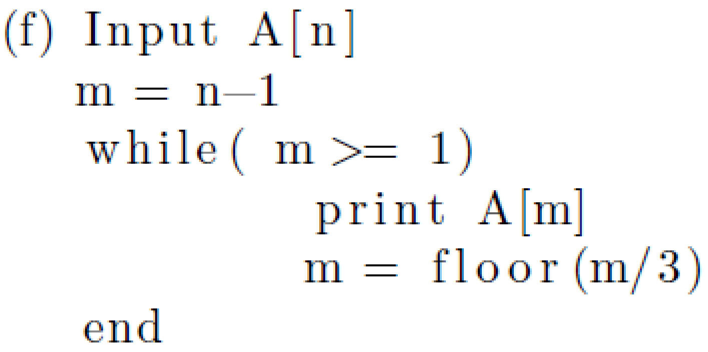
```
(b) Input A[n]
    c=0
    for i=1 to n*n
            for j=1 to n*n*n
                    c=c+1
                    A[c%n]=A[c%n]+1
            end
    end
```



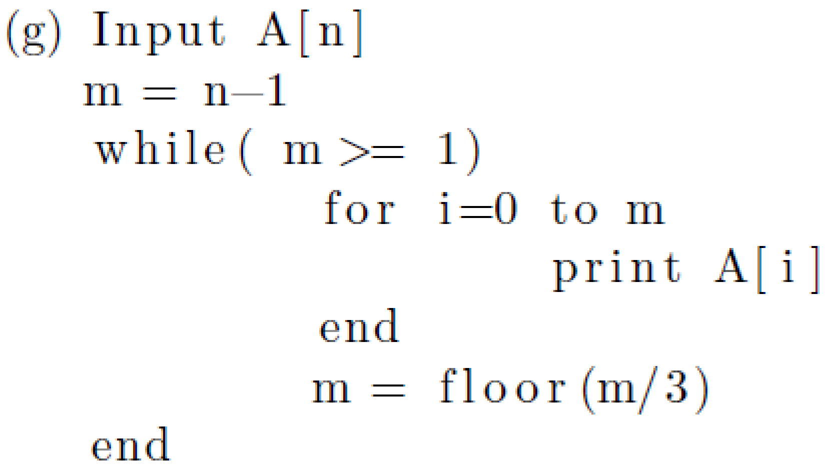
```
(c) Input A[n]
    c=0
    m=1
    for i=1 to n
            for j=1 to m*n
                    c=c+1
                    A[c%n]= A[c%n]+1
            end
            m = m/2;
    end
```



```
(e) Input A[n]
    m=1, c=0
    for i=1 to n
            for j=1 to m
                    c=c+1
            end
            m = m*2
    end
```



```
(f) Input A[n]
    m = n−1
    while( m >= 1)
                print A[m]
                m = floor(m/3)
    end
```



```
(g) Input A[n]
    m = n−1
    while( m >= 1)
            for i=0 to m
                    print A[i]
            end
            m = floor(m/3)
    end
```

Problem-9 Show that $\log^3 n$ is $o(n^{1/3})$.

Problem-10 Show that the summation $\sum_{i=1}^{n} \lceil \log_2 i \rceil$ is $\Omega(n \log n)$.

Thank You!!