Branch : CSE/IT **Batch : Hinglish**

WEEKLY TEST – 01

Subject : Algorithm

Topic : Analysis of Algorithm

Maximum Marks 15

Q.1 to 5 Carry ONE Mark Each

[MCQ]

1. sort the functions in increasing order of asymptotic (big – O) complexity

$f_1(n) = n^{0.999999}\log n$

$f_2(n) = 10000000n$

$f_3(n) = 1.000001^n$

$f_4(n) = n^2$

(a) $f_1(n), f_2(n), f_4(n), f_3(n)$

(b) $f_3(n), f_4(n), f_2(n), f_1(n)$

(c) $f_2(n), f_4(n), f_1(n), f_3(n)$

(d) None of these

[MCQ]

2. Sort the function in decreasing order of asymptotic (big – O) complexity: -

$f_1(n) = 2^{2^{1000000}}$

$f_2(n) = 2^{100000^n}$

$f_3(n) = n\sqrt{n}$

(a) $f_2(n), f_1(n), f_3(n)$ (b) $f_3(n), f_1(n), f_2(n)$

(c) $f_1(n), f_3(n), f_2(n)$ (d) None of these

[MCQ]

3. We know that O(g(n))

$$= \begin{cases} f(n): \text{There exist positive constant c and } n_0 \\ \text{Such that } 0 \le f(n) \le c.g(n), \text{for all } n \ge n_0 \end{cases}$$

Given that $f(n) = 10n^2 + 100$ and $g(n) = 2^n$;

where n and no are both positive integers.

if c = 0.125 then for which value of n_0,

f(n) = O(g(n))?

(a) 12 (b) 13

(c) 14 (d) 15

[MCQ]

4. Suppose that there are 3 programs X_1, X_2 and X_3 having time complexities $f_1(n)$, $f_2(n)$ and $f_3(n)$ respectively. Such that $f_1(n)$ is $O(f_2(n))$, $f_2(n)$ is $O(f_1(n))$, $f_1(n)$ is $O(f_3(n))$ and $f_3(n)$ is not $O(f_1(n))$. Then which one of the statements is true from the following statements?

(a) X_3 is always faster than X_1 and X_2 for very large size inputs

(b) X_1 is faster than X_2 and X_3 for very large inputs

(c) X_3 is slower than X_1 and X_2 for very large input

(d) X_2 is faster than X_1 and X_3 for very large size inputs

[MCQ]

5. Let $f(n) = \log\log\log\sqrt{n}$ and $g(n) = 2^{30^{30^{30}}}$ then which one of the following is true?

(i) $f(n) = \theta(g(n))$

(ii) $f(n) = \Omega(g(n))$

(iii) $f(n) = O(g(n))$

(iv) $f(n) = \omega(g(n))$

(a) (i), (ii) and (iii) only

(b) (ii), (iii) and (iv) only

(c) (ii) and (iv) only

(d) (iv) only

Q.5 to 10 Carry TWO Mark Each

[MCQ]

6. Consider the following code.

```
int a = 0;
for(int x = 0; x < n; x ++) {
      if (x%5==0){
            for (int y = 0; y < n; y ++){
                  if (x  = = y)
                        a+ = x * y )
                  }
      }
}
```

What is the highest asymptotic worst case time complexity of above code fragment?

(a) $O(n^2)$ (b) $O(\sqrt{n})$

(c) O(n) (d) O(log n)

[MCQ]

7. Arrange following function in the ascending order growth rate.

$f_1 = (1 + 0.0001)^n$, $f_2 = \sqrt{n}^{\log n}$, $f_3 = (1.005)^{\sqrt{n}}$

$f_4 = (\log n)^{\sqrt{n}}$, $f_5 = (\sqrt{n})^{\log n^2}$

(a) f_2, f_5, f_3, f_4, f_1 (b) f_3, f_4, f_2, f_5, f_1

(c) f_2, f_5, f_1, f_3, f_4 (d) None of the above

[MCQ]

8. What is the time complexity of the following code ?

```
for (a = 0; a < n – 2; a ++)
{
      for (b = 0; b < 100 ; b = b + 2)
         {
            for (c = 1; c < 8*n; c ++)
              {
                              ----------
                              ----------
                              ----------
                        }
                  }
      }
```

(a) $O(n^3)$ (b) $O(n^2)$

(c) O(logn) (d) O(n)

[NAT]

9. How many of the following statements is/are false____.

(i) $10\sqrt{n} + \log n = O(n)$

(ii) $\sqrt{n} + \log n = O(\log n)$

(iii) $\sqrt{n} + \log n = \theta(n)$

(iv) $\sqrt{n} + \log n = \theta(\sqrt{n})$

[MCQ]

10. Consider the following recursion function

```
P(n)
{
      if (n < = 0)
      return 1;
else if (n% 2 = = 0)
return P(n – 1);
else
return P(n–2);
}
```

What is the time complexity of above code?

(a) $\theta(\log n)$ (b) $\theta(2^n)$

(c) $\theta(n)$ (d) none of these

Answer Key

1. (a)
2. (d)
3. (d)
4. (c)
5. (c)
6. (a)
7. (a)
8. (b)
9. (2)
10. (c)

Hints and Solutions

1. (a)

The correct order of these functions is

$f_1(n)$, $f_2(n)$, $f_4(n)$, $f_3(n)$. To see why, $f_1(n)$ grows asymptotically slower than $f_2(n)$, recall that for any c > 0, log n is $O(n^c)$. Therefore, we have

$f_1(n) = n^{0.999999} \log n = 0(n^{0.999999}.n^{0.000001}) = O(n) = O(f_2(n))$

The function $f_2(n)$ is linear, while function $f_4(n)$ is quadratic, So $f_2(n)$ is $O(f_4(n))$. Finally, we know that $f_3(n)$ is exponential, which grows faster than quadratic, So, $f_4(n)$ is $O(f_3(n))$.

2. (d)

The correct order of these functions is $f_2(n)$, $f_3(n)$, $f_1(n)$ in decreasing order. The variable n never appears in the formula $f_1(n)$. So despite the multiple exponentials, $f_1(n)$ is constant. Hence it is asymptotically smaller than $f_3(n)$ which does grow with 'n'.

$f_2(n) = 2^{100000n}$ $\qquad$ $f_3(n) = n\sqrt{n}$

n = 1 $\qquad 2^{100000\times 1} > 1\sqrt{1}$

n = 2 $\quad 2^{100000\times 2} > 2\sqrt{2}$

$\vdots$

$\therefore f_2(n) > f_3(n)$

Option (a), (b), (c) are not correct option.

So, $f_2(n)$ $f_3(n)$ $f_1(n)$ is correct decreasing order.

Hence option (d) is correct.

3. (d)

Given C = 0.125

$f(n) \leq C.g(n)$

$10n^2 + 100 \leq 0.125 \times 2^n$

We need to check from n = 1 to 15

So, if n = 15

$10 * (15)^2 + 100 \leq 0.125 \times 2^{15}$

$2250 + 100 \leq 0.125 \times 32768$

$2350 \leq 4096$ (True)

4. (c)

Given, $\quad f_1(n) = O(f_2(n))$

$f_2(n) = O(f_1(n))$

$f_1(n) = O(f_3(n))$

$f_3(n) \neq O(f_1(n))$

The above functions conclude that, growth of f3 is larger than the growth rate of f_1 and f_2.

$\therefore \quad x_3$ is slower than x_1 or x_2.

5. (c)

$f(n) = \log\log\log\sqrt{n}$, $g(n) = 2^{20^{20^{20}}}$

$f(n) = \Omega(g(n)$ because g(n) is constant and

f(n) is depending on 2 ,therefore it is correct.

$f(n) \neq O(g(n))$ because f(n) is greater than g(n)

f(n) = w(g(n)) because f(n) > g(n) ,Correct.

$\therefore \quad$ (ii) and (iv) are true.

6. (a)

The inner most loop (if statement) executes per loop, we must check x = = y is true one per each iteration. This will take some non-zero constant amount of time. So the innermost loop will perform approximately n work.

The outer most loop and if statement will perform 'n' work during only $1/5^{th}$ of the iteration and will perform a constant amount of wort in the remaining $4/5^{th}$ of the time.

So, total amount of work done is approximately $\frac{n}{5}\cdot n + \frac{4n}{5}\cdot 1$

$\therefore T(n) = \frac{n^2}{5} + \frac{4n}{5}$

Which is $O(n^2)$

7. (a)

As we can see f_1 and f_3 are similar so lets compare these first.

$f_1 = (1 + 0.0001)^n$

$f_3 = (1.005)^{\sqrt{n}}$

$\therefore$ By seeing n and $\sqrt{n}$ in exponents we can conclude that $f_1 > f_3$ ….. (a)

Now comparing f_2, f_4, and f_5

$f_2 = \left(\sqrt{n}\right)^{\log n}$, taking log

$\log n \log \sqrt{n}$

$f_4 = (\log n)^{\sqrt{n}}$, taking log

$\sqrt{n}\log(\log n)$

$f_5 = \left(\sqrt{n}\right)^{\log n^2}$, taking log

$\log n^2 \log\left(\sqrt{n}\right)$

Form above we can clearly see that

$f_5 > f_2$ and $f_4 > f_5$ because of $\sqrt{n} > \log n$ ……. (b)

Now comparing f_1 and f_2

$(1 + 0.0001)^n = \left(\sqrt{n}\right)^{\log n}$

Taking log on both sides we get

$n \log 1.001 = \log n \log \sqrt{n}$

By comparing above we can clearly say that 'n' of f_1 will always be greater and n makes greater than f_2.

$\therefore$ $f_1 > f_2$ …..(c)

and similarly $f_5 > f_1$ …..(d)

also, $f_1 > f_4$ solving by using log….(e)

$\therefore$ from (a), (b), (c), (d) and (e)

We can conclude that

$f_1 > f_4 > f_3 > f_5 > f_2$

$\therefore$ $f_2\ f_5\ f_3\ f_4\ f_1$ is correct answer.

8. (b)

- a ranging from 0 t n – 2 in first loop so it is O(n)
- b is ranging from 0 to 100 which is constant time in 2nd (inner loop) so O(1)
- C is ranging from 1 to 8 n inner loop so it is O (n)

  n * 1 * n which is n^2

$\therefore$ $O(n^2)$, so option (b) is correct.

9. (2)

(i) $10\sqrt{n} + \log n = O(n)$ Correct, because

$\therefore$ $\sqrt{n} < n$

(ii) $\sqrt{n} + \log n = O(\log n)$ incorrect because

$\sqrt{n} > \log n$

(iii) $\sqrt{n} + \log n = \theta(n)$ incorrect it should be O(n) or θ(log)

(iv) $\sqrt{n} + \log n = \theta\sqrt{n}$, Correct

10. (c)

One of P(n – 1) of P(n – 2) will be called ,

In worst case

$T(n) = T(n – 1) + O(1)$

$T(n) = \theta(n)$