

- ① What do you understand by Asymptotic notations
 Define different Asymptotic notation with examples.

Ans Asymptotic notations are mathematical tools to represent time complexity of algorithms for asymptotic analysis.

- ① Θ Notation: $f(n) = \Theta(g(n))$

iff $0 <= c_1 * g(n) <= f(n) <= c_2 * g(n)$
 for all $n \geq n_0$.

Ex: $3n^3 + 6n^2 + 6000 = \Theta(n^3)$

- ② Big O Notation: $f(n) = O(g(n))$

iff $0 <= f(n) <= c * g(n)$
 for all $n \geq n_0$

Ex: $3n^3 + 10n + 32 = O(n^3)$

- ③ Ω Notation: $f(n) = \Omega(g(n))$

iff $0 <= c * g(n) <= f(n)$
 for all $n \geq n_0$

Ex: $n^2 + \log n = \Omega(n^2)$.

- ④ Small o Notation: $f(n) = o(g(n))$

iff $0 <= f(n) < c * g(n)$
 for all $n \geq n_0$

Ex: $3n^3 + 10n + 32 = o(n^4)$.

⑤ ω Notation : $f(n) = \omega(g(n))$

if, $0 <= c \cdot g(n) < f(n)$
for all $n > n_0$

Ex: $n^2 + n = \omega(n)$

② what should be time complexity of -
for $(i=1 \text{ to } n)$ & $i = i + 2$;

Ans The sequence is $2, 4, 8, \dots, 2^k$.
till $2^k = n$

$$k = \log_2 n$$

$$\therefore T(n) = O(\log_2 n)$$

③ $T(n) = \{3T(n-1) \text{ if } n > 0, \text{ otherwise } 1\}$

Ans using substitution.

$$\begin{aligned} T(n) &= 3T(n-1) \\ &= 3^2 T(n-2) \end{aligned}$$

\vdots

$$= 3^n T(0)$$

$$T(n) = O(3^n) \quad (\because T(0) = 1).$$

④ $T(n) = \{2T(n-1) - 1 \text{ if } n > 0, \text{ otherwise } 1\}$

Ans $T(n) = 2T(n-1) - 1$
using substitution,

$$T(n) = 2^2(T(n-2)) - 2 - 1$$

$$= 2^3(T(n-3)) - 2^2 - 2 - 2^0$$

$$= 2^n T(0) - 2^{n-1} - 2^{n-2} \dots - 2^1 - 2^0$$

$$= 2^n - (2^{n-1} + 2^{n-2} \dots + 2^0)$$

$$= 2^n - \left(\frac{2^n - 1}{2 - 1} \right)$$

$$= 1$$

$$\therefore T(n) = O(1).$$

⑤ what should be time complexity of:

int $i=1$, $s=1$;

while ($s \leq n$) {

$i++$; $s=s+i$;

printf("#");

}

Ans values of s are: ~~1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000~~

1, 1+2, 1+2+3, ... 1+2+3... K terms.

1+2+3... K terms $> n$.

$$\frac{K(K+1)}{2} > n.$$

$$\therefore K = O(\sqrt{n})$$

Hence, time complexity is $O(\sqrt{n})$.

⑥ Time complexity of:

void function (int n) {

int i, count = 0;

for (i=1; i+i <= n; i++)

count++;

}

Ans values to be checked.

$$1^2, 2^2, 3^2, \dots, K^2$$

$$K^2 = n$$

$$K = O(\sqrt{n}).$$

\therefore time complexity is $O(\sqrt{n})$.

④. Time complexity of
void function (int n) {

int i, j, k, count = 0;

for (i = n/2; i <= n; i++) $\rightarrow O(n)$

for (j = 1; j <= n; j = j * 2) $\rightarrow (O(\log_2 n))$

for (k = 1; k <= n; k = k * 2) $(O(\log_2 n))$.

count++

}

Ans

Outermost loop runs $n/2$ times.

for inner loop with variable j.

$$2^0, 2^1, 2^2, \dots, 2^K$$

$$2^K = n$$

$$K = \log_2 n$$

Similarly, loop with variable k runs $\log_2 n$ times.

\therefore time complexity is $O(n(\log_2 n)^2)$

8. Time Complexity of :

function (int n) {

if (n == 1) return;

for (i = 1 to n) {

for (j = 1 to n) {

print(" ");

}

}

function (n-3);

}

Ans The recursive function calls itself $n/3$ times. So, time complexity for that is $O(n)$.

Each recursive call has a nested loop with time complexity $O(n^2)$.

\therefore total time complexity is $O(n^3)$.

9. Time Complexity of :

void function (int n) {

for (i = 1 to n) {

for (j = 1; j <= n; j = j + 1)

print(" * ");

}

}

Ans total number of operations

$$= \begin{matrix} i=1 & j=2 & i=3 \\ \text{1 times} & + \frac{n}{2} \text{ times} & + \frac{n}{3} \text{ times} \dots \dots \dots \end{matrix} \quad \text{+ time}$$

$$= 1 + \frac{n}{n-1} \dots \dots \dots \frac{n}{2} + \frac{n}{1} \text{ times.}$$

$$= n \left(\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n} \right)$$

$$= n \log n$$

$$= O(n \log n) \rightarrow \text{time complexity.}$$

⑩. for the function, n^k & c^n what is the asymptotic relationship between these functions? Assume that $k > 1$ & $c > 1$ are constants. find out value of c & n_0 for which relation holds.

Ans Exponent function grows the fastest hence.

$$n^k \text{ is } O(c^n)$$

$$\text{let, } k = 3 \text{ \& } c = 2.$$

for $n > 10$, the above relation holds.

Similarly, there can be infinite number of pair c & n_0 for which above relation holds.