

GATE - 2007
INFORMATION TECHNOLOGY
DETAILED SOLUTIONS TO SOME OF THE
PROBLEMS



SOLUTIONS TO ABOUT 70
MARKS

THEORY OF COMPUTATION
PROGRAMMING
NUMERICAL METHODS
MATEMATICS

SOFTWARE ENGINEERING
DISCRETE MATHEMATICS
COMPUTER ORGANISATION
DATA STRUCTURES

DESIGN & ANALYSIS OF ALGORITHMS
OPERATING SYSTEMS

TIME TAKEN: 10 HOURS TO SOLVE AND
ENTER THE SOLUTIONS

QUESTION NUMBER	AREA	ANSWER	QUESTION NUMBER	AREA	ANSWER
1	DM		2	NM	
3	DAA	D	4	SE	D
5	SE	D	6	CA	
7	CO		8	CO	
9	TOC	C	10	OS/P	C
11	CA		12	OS	
13	CN/WT		14	CN/WT	
15	DS/CN		16	M	D
17	DAA	A	18	CN/TOC	D
19	WT		20	WT	
21	DM		22	NM	
23	DM	A	24	DAA	B
25	DAA	D	26	DAA	
27	P	D	28	DAA/DM	
29	DAA/DS	D	30	DAA	B
31	P	C	32	DS/P	B
33	P	D	34	P	A
35	P/PL	D	36	CA	
37	CA		38	CO	
39	DBMS		40	CO	
41	CA		42	CO/CA/DC	
43	CO/DC		44	CA	
45	CA		46	TOC	A
47	TOC	D	48	TOC	C
49	TOC	C	50	TOC	A
51	SE	C	52	SE	B
53	SE	A	54	SE	D
55	SE	B	56	OS/SE	D
57	OS		58	OS	
59	UNIX		60	CN	
61	DC/CN		62	DC/CN	
63	CN		64	CN	
65	DBMS/DM		66	P/SE/DBMS	A

67	DBMS		68	DBMS	
69	DBMS		70	DC	
71	TOC	A	72	TOC	A
73	TOC	C	74	CN	
75	CN		76	DM/DAA	A
77	DAA/DM	D	78	CO	
79	CO		80	DAA	
81	DAA		82	CA	
83	CA		84	DBMS/DS	
85	DBMS/DS				

Q3.[DAA] Consider a weighted undirected graph with positive edge weights and let uv be an edge in the graph, It is known that the shortest path from the source vertex s to u has weight 53 and the shortest path from s to v has weight 65. Which one of the following statements is always true?

- (A) $\text{weight}(u,v) < 12$
- (B) $\text{weight}(u,v) \leq 12$
- (C) $\text{weight}(u,v) > 12$
- (D) $\text{weight}(u,v) \geq 12$

Answer: (D)

Explanation: If the $\text{weight}(u,v) < 12$ then there is a shorter path then 65 to v from s . Simply go to u and then to v . So this rules out (A) and (B). One can design a graph where (u,v) is on the shortest path so this rules out (C). The answer is (D).



Q4. [SOFTWARE ENGINEERING]

In the Spiral model of software development, the primary determinant in selecting activities in each iteration is

- (A) Iteration size
- (B) Cost
- (C) Adopted process such as Rational Unified Process or Extreme Programming
- (E) Risk

Answer: (E)

Explanation: The Spiral model is basically a risk driven mode. The number of iterations is not fixed. Though the spiral method is normally applied to large projects it is not the cost that is the main consideration for the choice of the model.



Q5.[Which of the following systems is a most likely candidate of a pipe and filter architecture?

- (A) Expert system
- (B) DB repository
- (C) Aircraft flight controller
- (D) Signal processing

Answer: (D).

Explanation: As the stream of signal flows it can be computed upon.



Q9.[TOC]

Consider an ambiguous grammar G and its disambiguated version D. Let the language recognized by the two grammars be denoted by $L(G)$ and $L(D)$ respectively. Which one of the following is true?

- (A) $L(D) \subset L(G)$
- (B) $L(D) \supset L(G)$
- (C) $L(G) = L(D)$
- (D) $L(D)$ is empty

Answer: (C)

Explanation: The grammar at most changes but the language generated remains the same.



Q10.[P/SOFTWARE ENGINEERING/OS]

Processes P1 and P2 use `critical_flag` in the following routine to achieve mutual exclusion. Assume that `critical_flag` is initialized to `FALSE` in the main program.

```
Get_exclusive access()  
{  
    if (critical_flag ==FALSE) {  
        critical_flag = TRUE;  
        critical_region();  
        critical_flag = FALSE;  
    }  
}
```

Consider the following statements.

- (i) It is possible for both P1 and P2 to access `critical_region` concurrently.**
- (ii) This may lead to a deadlock.**

Which of the following holds?

- (A) (i) is false and (ii) is true**
- (B) Both (i) and (ii) are false**
- (C) (i) is true and (ii) is false**
- (D) Both (i) and (ii) are true**

Answer: (C)

Explanation: As the check for `critical_flag` being `FALSE` is not done as a single undivided operation both the processes can be in the critical region. There is no competition for more than one resource, so there will be no deadlock. So (i) is true and (ii) is false. This gives choice (C).



Q16. [MATHS]

The minimum positive integer p such that 3^p modulo $17 = 1$ is

(A) 5 (B) 8 (C) 12 (D) 16

Answer: (D)

Explanation: By Fermat's theorem, 3^{17-1} modulo $17 = 1$. So $p=16$ is the solution.



Q17. Exponentiation is a heavily used operation in public key cryptography. Which of the following options is the tightest upper bound on the number of multiplications required to compute b^n modulo m , $0 < b, n < m$?

(A) $O(\log n)$ (B) $O(\sqrt{n})$ (C) $O(n / \log n)$ (D) $O(n)$

Answer: (A)

Explanation: One can compute b, b^2, b^4, b^8, \dots and combine them to the extent possible. This means a continuous division of n by 2 and so give a complexity of $O(\log n)$.



Q18. [THEORY OF COMPUTATION] A firewall is to be configured to allow hosts in a private network to freely open TCP connections and send packets on open connections. However, it will only allow external hosts to send packets on existing open TCP connections or connections that are being opened (by internal hosts) but not allow them to open

TCP connections to hosts in the private network. To achieve this the minimum capability of the firewall should be that of

- (A) A combinational circuit**
- (B) A finite automaton**
- (C) A pushdown automata with one stack**
- (D) A pushdown automata with two stacks**

Answer: D

Explanation: A pushdown automata with two stacks can simulate a general purpose computer. The number of internal hosts and external hosts is not bounded, and we need comparison so (A) and (B) cannot help as they deal with bounded numbers.



Q23. [DISCRETE MATHEMATICS]

A partial order P is defined on the set of natural numbers as follows. Here x/y denotes integer division.

Consider the following ordered pairs:

- (i) (101,22) (ii) (22,101) (iii) (145,265) (iv) (0,153)**

Which of these ordered pairs of natural numbers are contained in P?

- (A) (i) and (ii) (B) (i) and (iii) (C) (i) and (iv) (D) (iii) and (iv)**

Answer: (B)

Explanation: Compute the residues for the four pairs:

- (i) (1,2) (ii) (2,1) (iii) (5,5) (iv) (0,3)**

In an ordered pair (a,b) we required $a < b$ where a and b are the residues.

Only (i) and (iii) satisfy the requirement. So answer is (B). We exclude (iv) as (0,0) ordered pair is specifically given.



Q24.[DAA]

A depth first search is performed on a directed acyclic graph. Let $d[u]$ denote the time at which vertex u is visited for the first time and $f[u]$ the time at which the dfs call to the vertex terminates. Which of the following statements is always true for all edges (u,v) in the graph?

- (A) $d[u] < d[v]$ (B) $d[u] < f[v]$ (C) $f[u] < f[v]$ (D) $f[u] > f[v]$

Answer: (B)

Explanation: For a forward edge (u,v) we have $f[u] < f[v]$ not correct.

For a back edge we have $f[u] > f[v]$ not correct.

For a back edge we have $d[u] > d[v]$ so the answer is (B).



Q25.[DAA]

Which is the largest integer m such that every simple connected graph with n vertices and n edges contains at least m different spanning trees?

- (A) 1 (B) 2 (C) 3 (D) n

Answer: (D)

Explanation: For a graph that looks like a triangle we have three vertices, and three spanning trees.

For a graph that looks like a rectangle we have four vertices and four spanning trees.

For a polygon of n vertices, n edges we have n spanning trees by eliminating any edge.

So answer is n i.e.(D).



Q27. [P]

The function f is defined as follows:

```

int f(int n) {
    if (n <= 1) return 1;
    else if (n % 2 == 0) return f(n/2);
    else return f(3n - 1);
}

```

Assuming that arbitrarily large integers can be passed as a parameter to the function, consider the following statements.

- (i) The function f terminates for finitely many different values of $n \geq 1$.
- (ii) The function f terminates for infinitely many different values of $n \geq 1$.
- (iii) The function f does not terminate for finitely any different values of $n \geq 1$.
- (iv) The function f does not terminate for infinitely many different values of $n \geq 1$.

Which one of the following options is true of the above?

- (A) (i) and (iii)
- (B) (i) and (iv)
- (C) (ii) and (iii)
- (D) (ii) and (iv)

Answer: (D)

Explanation: If n is a power of 2 then the function $f(n/2)$ is continuously executed till $f(1)$ is called. So (ii) is true and (i) is not true if we take that it means it terminates only for a finite number of values of n .

If n is not a power of 2, then $f(3n-1)$ is called. Let $3n-1 = 2^k$, this solves to $n = (2^k + 1)/3$ which has an infinite number of values of k satisfying it. So (iv) is correct.

Combining both the statements, (i) and (iv) are correct, i.e. answer is (D).



Q29.[DATA STRUCTURES/DAA]

When searching for the key value 60 in a binary search tree, nodes containing the key values 10, 20, 40, 50, 70, 80, 90 are traversed, not

necessarily in the order given. How many different orders are possible in which these key values can occur on the search path from the root to the node containing the value 60?

(A) 35 (B) 64 (C) 128 (D) 5040

Answer: (D)

Explanation; The seven elements of the key values can come in 7! different orders when the tree is to be set up. This is 5040 ways.



Q30.[DATA STRUCTURES]

Suppose you are given an implementation of a queue of integers. The operations that can be performed on the queue are:

- (i) isEmpty(Q) ---- returns true if the queue is empty, false otherwise.
- (ii) delete(Q) -----deletes the element at the front of the queue and returns its value.
- (iii) insert(Q)---inserts the integer I at the rear of the queue.

Consider the following function:

```
void f (queue Q) {  
    int I;  
    if (!isEmpty(Q)) {  
        I = delete(Q);  
        f(Q);  
        insert(Q,I);  
    }  
}
```

What operation is performed by the above function f?

- (A) Leaves the queue Q unchanged.
- (B) Reverses the order of the elements in the queue Q.
- (C) Deletes the element at the front of the queue Q and inserts it at the rear keeping the other elements in the same order.
- (D) Empties the queue Q.

Answer: (B)

Explanation: Consider a queue consisting of two elements 1 and 2. Two calls are made and in the first call 1 is deleted, and in the 2nd call 2 is deleted. Then they are inserted 2 first when f(Q) returns and then 1.



Q31[PROGRAMMING]

Consider the C program given below:

```
#include <stdio.h>
int main() {
    int sum = 0, maxsum = 0, I, n = 6;
    int a[] = {2, -2, -1, 3, 4, 2};
    for (I = 0; I < n; I++) {
        if(I == 0 || a[I] < 0 || a[I] < a[I-1]) {
            if(sum > maxsum) maxsum = sum;
            sum = (a[I] > 0) ? a[I] : 0;
        }
        else sum += a[I];
    }
    if (sum > maxsum) maxsum = sum;
    printf ("%d\n", maxsum);
}
```

What is the value printed out when this program is executed?

- (A) 9 (B) 8 (C) 7 (D) 6

Answer: (C)

Explanation: See the simulation below:

sum	maxsum	I
0	0	0
0	0	1
0	0	2
3	0	3
7	0	4
2	7	5
2	7	



Q32. [PROGRAMMING]

Consider the following C program:

```
#include <stdio.h>
#define EOF -1
void push(int); /* push the argument on the stack */
int pop(void); /* pop the top of the stack */
void flagError();

int main()
{
    int c, m, n, r;

    while ((c = getchar()) != EOF)
    {
        if (isdigit(c) push(c);
        else if ( (c == '+') || (c == '*'))
        {
            m = pop();
            n = pop();
            r = (c == '+') ? n+m : n*m;
            push(r);
        }
        else if (c != ' ') flagError();
    }
    printf ("%c", pop());
}
```

What is the output of the program for the following input?

5 2 * 3 3 2 + * +

- (A) 15 (B) 25 (C) 30 (D) 150

Answer: (B)

Explanation: See the simulation of the stack below.

Incoming symbol is 5, push it on the stack:

5

Incoming symbol is 2, push it on the stack:

2
5

Incoming symbol is *, pop the two top stack elements and replace by product.

2
5

Incoming symbol is *, pop the two top stack elements and replace by product.

10

Incoming symbol is 3, push it on the stack.

3
10

Incoming symbol is 3, push it on the stack.

3
3
10

Incoming symbol is 2, push it on the stack.

2
3
3
10

Incoming symbol is +, pop the two top stack elements and replace by their sum.

2
3
3
10

Incoming symbol is +, pop the two top stack elements and replace by their sum.

5
3

10

Incoming symbol is *, pop the two top stack elements and replace by their product.

15
10

Incoming symbol is +, pop the two top stack elements and replace by their sum.

25

Pop the stack and print the value 25.



Q13. [PROGRAMMING]

Consider the program below in a hypothetical language which allows global variables and a choice of call by reference or call by value methods of parameter passing.

Int I;

```
Program main()  
{    int j=60;
```

```
(i)   I=50;
(ii)  Call f(I,j);
      print(I,j);
}
```

```
procedure f(x,y)
{
  I = 100;
  x = 10;
  y = y + I;
}
```

Which one of the following options represents the correct output of the program for the two parameter passing mechanisms?

- (A) Call by value: I=70, j=10; Call by reference: I=60, j=70;
- (B) Call by value: I=50, j=60; Call by reference: I=50, j=70;
- (C) Call by value: I=10, j=70; Call by reference: I=100, j=60;
- (D) Call by value: I=100, j=60; Call by reference: I=10, j=70;

Answer: (D)

Explanation: (QUICK) In call by value the calling parameter does not change so the I in the procedure is the global I which is assigned 100.

Only (D) satisfies the answer.

(LONG) In procedure f when x and y are called by value, whatever we do to x and y in the procedure will not alter the actual parameters. The global I is given the value of 100 and when we return to the main program j retains its value of 60.

In the case of call by reference x refers to I the global variable, which is assigned the value 100 first and then 10. y corresponds to j, which has an initial value of 60 when we enter the procedure and its value is augmented by 10, giving a final value of 70.



Q34.[PROGRAMMING]

Consider the program below in a hypothetical programming language which allows global variables and a choice of static or dynamic binding.

Int I;

```
Program main()  
{   I = 10;  
    call f();  
}  
procedure f()  
{   int I=20;  
    call g();  
}  
procedure g()  
{   print I;  
}
```

Let x be the value printed under static scoping and y be the value printed under dynamic scoping. Then, x and y are

- (A) $x=10, y=20$ (B) $x=20, y=10$
(C) $x=10, y=10$ (D) $x=20, y=20$

Answer: (A)

Explanation: Under static scoping the I in the print command refers to the global I which is set to 10 in the main program, and the assignment to I in the procedure f is to a local variable I , so the value printed for static scoping x must be 10.

Under dynamic scoping, the I in the print command, tracing the chain of calls is the I referred to in f , and so the value printed is 10.

So $x=10, y=10$ is printed and the answer is (A).



Q25.[PROGRAMMING]

Early binding refers to a binding performed at compile time and late binding refers to a binding performed at execution time. Consider the following statements:

- (i) Static scope facilitates w1 bindings.
- (ii) Dynamic scope requires w2 bindings.
- (iii) Early bindings w3 execution efficiency.
- (iv) Late binding w4 execution efficiency.

The right choices of w_1 , w_2 , w_3 and w_4 (in that order) are

- (A) Early, late, decrease, increase
- (B) Late, early, increase, decrease
- (C) Late, early, decrease, increase
- (D) Early, late, increase, decrease

Answer: (D)

Answer: Early binding by static scope can be computed by examining the program, and the code can be optimized. Dynamic scope can only be determined at runtime and so w_2 has to be done late. Static scope, w_1 bindings can be optimized and increase execution efficiency. Late binding w_4 , eat up execution time and thus decrease execution efficiency. So the answer is (D).



Statement for Linked answer questions 76 & &

Consider the sequence $\{x_n\}_{n \geq 0}$ defined by the recurrence relation $x_{n+1} = cx_n^2 - 2$, where $c > 0$.

Q76. [DAA/DISCRETE MATHS]

Suppose there exists a non-empty open interval (a, b) such that for all x_0 satisfying $a < x_n < b$, the sequence converges to a limit. The sequence converges to the value

- (A) $(1 + \sqrt{1+8c})/(2c)$
- (B) $(1 - \sqrt{1+8c})/(2c)$
- (C) 2
- (D) $2/(2c-1)$

Q77. For which of the following values of c , does there exist a non-empty open interval (a, b) such that the sequence x_n converges for all x_0 satisfying $a < x_0 < b$?

- (i) 0.25 (ii) 0.35 (iii) 0.45 (iv) 0.5
- (A) (i) only
- (B) (i) and (ii) only
- (C) (i), (ii) and (iii) only

(D) (i), (ii), (iii) and (iv)

Q76. Answer: (A)

Q77. Answer: (D)

Explanation:

The recurrence relation is $x(n+1) = cx(n)^2 - 2$

So $cx(n)^2 = 2 + x(n+1)$

$x(n) = \sqrt{[(2/c) + x(n+1)/c]}$

$= \sqrt{[(2/c) + 1/c\sqrt{[(2/c) + x(n+2)/c]]}]}$

$= \sqrt{[(2/c) + 1/c\sqrt{[(2/c) + 1/c\sqrt{[(2/c) + x(n+3)/c]]}]]}]}$

Squaring both sides

$x(n)^2 = 2/c + (1/c)x(n)$

Solving the equation for the positive root

$x(n) = (1 + \sqrt{[1+8c]})/(2c)$

Or the answer is (A) for Q76.

Consider different values of c in the root $x(n) = (1 + \sqrt{[1+8c]})/(2c)$

- (i) c = 0.25 then xn is 5,46**
- (ii) c = 0.35 then xn is 4.14**
- (iii) c = 0.45 then xn is 3.44**
- (iv) c = 0.5 then xn is 3.3**

For the series to converge the limit and n tends to infinity of $x(n+1)/x(n)$ should be <1 . From the recurrence we should have $cx(n)^2 - x(n) - 2 < 0$

For all the above values of c we have the lhs negative so all the c values are acceptable and the answer is (D).



