Ranker Test (as per GATE Pattern)

Subjects : (1) C-Programming
(2) Digital Logic
(3) Theory of Computation
(4) Discrete Mathematics

[NAT 1 Mark]

1. Consider the following program:

```
#include<stdio.h>
int r()
  {
    static int i;
    return i++;
  }
int main()
  {
    int i=0, j=1;
    for(i++; i++<9; i++)
     {
       switch(r() + i)
        {
         case 1: j+= 2;
         break;
         default: j+= 3;
         case 2: j+= 1;
         break;
        }
     }
  }
```

The final value of i + j is ________.

[MCQ 1 Mark]

2. Consider the following two languages L_1 and L_2:

$L_1 = \{0^m1^n \mid m, n \geq 0\}$

$L_2 = \{0^m1^m \mid m \geq 0\}$

If $L = (L_1 \cap \overline{L_2})$ then the language L will be

(a) $L = \{0^m1^m \mid m \geq 0\}$

(b) $L = \{0^m1^n \mid m! = n\}$

(c) $L = \{X - 0^m1^n \mid X = \{0 + 1\}^*, m! = n\}$

(d) None of these.

[NAT 1 Mark]

3. Function $f = \overline{A}BD + \overline{A}CD + \overline{A}C\overline{D} + AB\overline{C}D + ABCD$ minimum number of NAND gates required to implement the function?

[NAT 1 Mark]

4. Consider a language $L = \{w \mid w \in \{a, b\}^*$, 8^{th} symbol from end is 'a'$\}$

If number of states in NFA are A and number of states in DFA are B then the value of A + B is ____.

[NAT 1 Mark]

5. Let $G = (V, E)$ be an undirected connected loop-free graph. suppose further that G is planar and determines 53 regions. If, for some planar embedding of G, each region has at least five edges in its boundary, then minimum number of vertices it can have _______.

[MCQ 1 Mark]

6. Consider the following program:

```
#include <stdio.h>
void func(char **p)
{
  printf("%s", *(p+3));
  printf("%s", *p++);
  printf("%s", *(p+++2));
  printf("%c", **p);
}
void main(){
char *a[ ]={"Parakram","2024","Shreshth","2025"};
func(a);
}
```

The output is___.

(a) 20252024reshthS

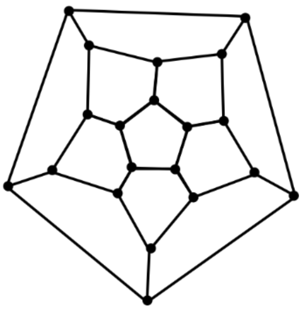
(b) 2025Parakram2025S

(c) 2025arakram2024S

(d) Compilation Error

[MSQ 1Mark]

7. Consider the following graph, and choose the correct statements.



(a) The given graph is Hamiltonian circuit.
(b) The chromatic number X(G) of the graph is 3.
(c) The given graph is an Eulerian.
(d) All the above statements are false.

[MSQ 1 Mark]

8. Which of the following statements is/are correct?
(a) The number 11101.11 on base 2 is equivalent to 1 D.3 on base 16.
(b) The number 1001.1 on base 2 is equivalent to 11.4 on base 8
(c) The number 112 on base 4 is equivalent to 211 on base 3
(d) The number 214 on base 5 is equivalent to 3 B on base 16.

[MCQ 1 Mark]

9. Which of the following is Turing acceptable language?
(a) Set of real numbers.
(b) Set of real numbers between 0 and 12.
(c) Set of prime numbers.
(d) All of the above.

[MCQ 1Mark]

10. A logical function $f(A, B, C) = \bar{A}B+\bar{B}C$.
Then the logical function $f_1(A,B,C)=f(f,\bar{f},A)$
Then $f_1(A,B,C)$ will be ?
(a) $AB+\bar{B}\bar{C}$ (b) $AB + BC$
(c) $A\bar{B}+B\bar{C}$ (d) $(A+\bar{B})(A+\bar{C})$

[MSQ 2 Marks]

11. Consider the following degree sequence:
5, p, 4, 4, 3, q, 2, 1 provided the sequence is graphical and the number of edges for the graph is 13, then find the possible values of p and q.
(a) 5, 2 (b) 7, 0
(c) 4, 3 (d) 6, 1

[MCQ 2 Marks]

12. Which of the following is undecidable?
(a) L = {< T > | T is Turing machine and it halts on some| string | $\le$ 200}.
(b) L = {< G > | G is CFG and L(G) $\neq \phi$}.
(c) L = {< L_1, L_2> | L_1 and L_2 are DCFL and $L_1 = L_2$}.
(d) None of these.

[NAT 2 Marks]

13.
```
#include <stdio.h>
void fun2(int n);
void fun1(int n)
{
  if(n < 2) return;
  fun2(n – 2);
  printf("%d\t", n – 2);
}
void fun2(int n)
{
  if(n < 1) return;
  printf("%d\t", n – 1);
  fun1(n – 1);
}
int main()
{
  fun1(5);
  fun2(5);
  return 0;
}
```
The sum of the printed values is ____________.

[NAT 2 Marks]

14. The initial state of mod-16 down counter is 0110. After 68 clock pulses, the state of the counter will be equivalent to decimal number?

[MCQ 2 Marks]

15. Consider the following grammars G_1, G_2 and G_3:
G_1 : $S \rightarrow P\ Q$
$P \rightarrow 0\ P\ 1 | \in$
$Q \rightarrow 1\ Q\ 2 | \in$
G_2 : $S \rightarrow 0\ S1 | Q$
$P \rightarrow 1Q\ 2 | \in$
G_3 : $S \rightarrow P\ Q\ |\ Q$
$P \rightarrow 0\ P\ 1 |\ 01$
$Q \rightarrow 1\ Q\ 2 | \in$
Here, {S,P,Q} are variables where S is start symbol. {0,1,2} are terminals.

Which of the following is true?
(a) G_1 and G_2 are equivalent.
(b) G_1 and G_3 are equivalent.
(c) G_2 and G_3 are equivalent.
(d) None of these.

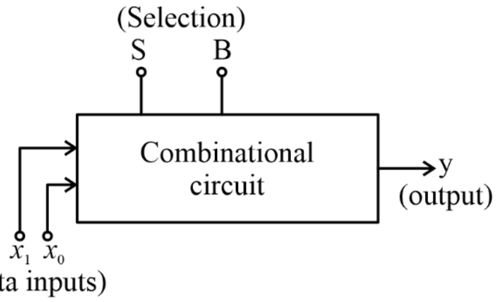
[MCQ 2 Marks]

16.
```
#include <stdio.h>
void fun1 (int n);
void fun2(int n);
void fun1(int n)
{
    if (n < 2) return;
    fun2(n – 2);
    printf("%d \t", n – 2);
}
void fun2 (int n)
{
    if (n < 1) return;
    printf("%d\t", n – 1);
    fun1(n – 1);
}
int main()
{
  fun1(5);
  fun2(5);
  return 0;
}
```
The sum of the printed values is ____________.

[MCQ 2 Marks]

17. The design of a combinational circuit is attempted as demonstrated below.

(i) For S = 0, y = 0 regardless of status of B
(ii) For S = 1 and B = 0, y = x_1
(iii) For S = 1 and B = 1, y = x_0



The minimum number count of 3-input NAND gates required to complete the design, will be?
(a) 3 (b) 2
(c) 4 (d) 1

[MCQ 2 Marks]

18. The output of the following C program is_____.
```
#include <stdio.h>
int main()
{
    void *p;
    char *c="GATEWallah";
    int a = 513;
    p = &a;
    printf("%d ", *(char*)p);
    p = c;
    printf((char*)p + 4);
    return 0;
}
```
(a) Garbage value
(b) 256Wallah
(c) 1Wallah
(d) 513EEWallah

[MSQ 2 Marks]

19. Consider the following languages:
L_1 = Context Free Language.
L_2 = Deterministic Context Free Language.
L_3 = Recursive Enumerable Language.
Which of the following is/are incorrect?
(a) $L_1 \oplus L_2$ is Context Free language but not Deterministic Context Free language.
(b) $L_1 \cup \overline{L_3}$ Recursive.
(c) $L_2 \cup \overline{L_3}$ is Recursive Enumerable.
(d) $\left(\overline{L_1} \cup \overline{L_2} \cup L_3\right)$ is Recursive Enumerable.

[MCQ 2 Marks]

20. Consider the following program:
```
#include<stdio.h>

int func(int i)

{

 i –= 3;

 return i;

}

void main(){

 int i = printf ("Parakram 2024");

 i = func (i = func (i = func(– –i)));

 printf("%d", i);

}
```
The output is-

(a) Parakram 20243
(b) Parakram 20246
(c) Parakram 20249
(d) Parakram 2024

[NAT 2 Marks]

21. How many don't care inputs are there in a BCD adder?

[MSQ 2 Marks]

22. Consider the following regular expressions P, Q and R over $\Sigma = \{a, b\}$:

P = ab + aQ + bR

Q = baQ + bR

R = Raba + a

Which of the following regular expression will produce all the strings accepted by above regular expression?

(a) $ab + ba(aba)^* [\in + a(ba)^*]$

(b) $ab + [\in + a(ba)^*] ba(aba)^*$

(c) $ab + a(ba)^+ ba(aba)^*$

(d) $ab + a(ba)^+ (aba)^* + ba(aba)^*$

[MCQ 2 Marks]

23. G be a connected graph in which only one node has degree > '1' and rest of the nodes are of degree 1. Add an edge between every two nodes of degree 1 in such a way that if a, b, c, d are node then a to b one edge, b to c one edge & d to a one edge. The resultant graph is sure to be

(a) Regular (b) Complete
(c) Hamiltonian (d) Euler

[NAT 2 Marks]

24. Consider the following program:

```
#include <stdio.h>
void func()          {
 int x = -5;
 static int y;
     do
     {
         x++;
         y++;
      } while (x ++< 1);
printf("%d\t %d\t", x, y);
}
void main()
{
   func();
   func();
}
```

The sum of the printed values is ________.

[NAT 2 Marks]

25. Let $L_1 = aa^* b^+$

$L_2 = ab\,(ab)^*$

$L_3 = L_1 / L_2^*$

The minimal number of states are needed for L_3 in DFA is___.

[MSQ 2 Marks]

26. A ripple counter is required to count from 0 to 255 in decimal with input clock frequency of 512 kHz.

(a) The mod of counter is 8.

(b) The frequency of the output of the eighth FF will be 2 kHz.

(c) The counter starts at 0000 0000. The state of counter after 520 clock pulses will be 0000 1000.

(d) The mod of counter is 256

[MCQ 2 Marks]

27. Consider the following program:

```
#include<stdio.h>
char *p[ ]={"GATE", "Wallah", "2024", "2025"};
char **q[ ]={p, p+3, p+1, p+2};
char ***r = q;
void main()
{
    printf("%s ", **++r);
    printf("%s ", *--*++r+2);
    printf("%s ", *r[-1]+3);
    printf("%s ", r[-1][-1]+1);
}
```

The output is-

(a) 2025 TE 4 024

(b) 2024 Wallah 5 025

(c) 2025 Wallah 5 024

(d) 2025 TE 5 024

[NAT 2Marks]

28. Consider the following statements:

S_1: For given context free language L, checking L is regular.

S_2: For given regular language L, checking L is CFL.

S_3: For given deterministic pushdown automata 'P', non-deterministic push down automata ‘N’ is equivalent to ‘P’.

S_4: Determine whether a given type-3 grammar is ambiguous.

Total number of undecidable statements are_____.

[MCQ 2 Marks]

29. Consider the following program:

```
#include <stdio.h>
void f()
{
 static int a = 3;
 int b = 5;
 a – = b++;
 printf("%d\t%d\n", a, b);
 }
int main()
{
     static int a = 2;
     int b = 1;
     f();
     a += 3;
     f();
     printf ("%d\t%d", a, b);
   return 0;
}
```

The output is-

(a) $\begin{matrix} -2 & 6 \\ -7 & 6 \\ 5 & 1 \end{matrix}$

(b) $\begin{matrix} -2 & 6 \\ -7 & 7 \\ 5 & 1 \end{matrix}$

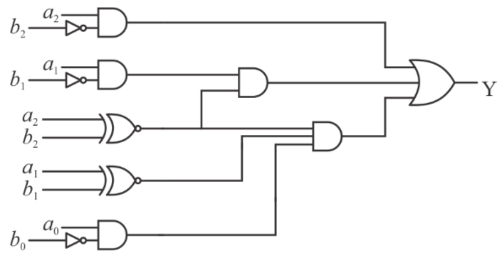
(c) $\begin{matrix} -2 & 5 \\ -7 & 6 \\ 5 & 1 \end{matrix}$

(d) None

30. [MCQ 2Marks]

A comparator circuit is designed to compare two modes A and B.

$A = a_2\, a_1\, a_0 \qquad B = b_2\, b_1\, b_0$



Then which of the following is true ?

(a) This circuit compares A and B and output Y is ‘1’ when A = B.

(b) This circuit compares A and B and output Y is ‘1’ when A > B.

(c) This circuit compares A and B and output Y is ‘1’ when B > A.

(d) This circuit gives the final carry output of (A+B) addition.

Answer Key

1.	**(24)**	**12.**	**(a)**	**23.**	**(c)**
2.	**(b)**	**13.**	**(12)**	**24.**	**(18)**
3.	**(4)**	**14.**	**(2)**	**25.**	**(4)**
4.	**(265)**	**15.**	**(b)**	**26.**	**(b, c, d)**
5.	**(82)**	**16.**	**(12)**	**27.**	**(d)**
6.	**(b)**	**17.**	**(c)**	**28.**	**(1)**
7.	**(a, b)**	**18.**	**(c)**	**29.**	**(a)**
8.	**(b, c, d)**	**19.**	**(a, b, c)**	**30.**	**(b)**
9.	**(c)**	**20.**	**(a)**		
10.	**(d)**	**21.**	**(312)**		
11.	**(a, c, d)**	**22.**	**(b, d)**		

Hints & Solutions

1. (24)

Initialization:
i++ or i=1;
Condition: i++<9 or 1<9 TRUE. i is incremented to 2.
Switch(r() + i) is equivalent to switch(0+2):
case 2 is executed: j=2;
i++; i = 3
Condition: i++<9 or 3<9 TRUE. i is incremented to 4.
Switch(r() + i) is equivalent to switch(1+4):
default and case 2 are executed: j=6;
i++; i=5
Condition: i++<9 or 5<9 TRUE. i is incremented to 6.
Switch(r() + i) is equivalent to switch(2+6):
default and case 2 are executed: j=10;
i++; i=7
Condition: i++<9 or 7<9 TRUE. i is incremented to 8.
Switch(r() + i) is equivalent to switch(3+6):
default and case 2 are executed: j=14;
i++; i = 9
Condition: i++<9 or 9<9 FALSE. i is incremented to 10.
Hence, i + j = 24.

2. (b)

- $L_1 = \{0^m 1^n \mid m,n \geq 1\}$

  regular expression = (0*1*)

- $L_2 = \{0^m 1^m \mid m \geq 0\}$ CFL

  $\overline{L_2} = (0 + 1)^* - \{0^m 1^m \mid m \geq 0\}$

  $L = L_1 \cap \overline{L_2}$

  $L = 0^*1^* \cap \overline{\{0^m 1^n \mid m \,!= n\}}$

  $L = \{0^m 1^n \mid m! = n\}$

Hence, option (b) is correct.

3. (4)

Given function

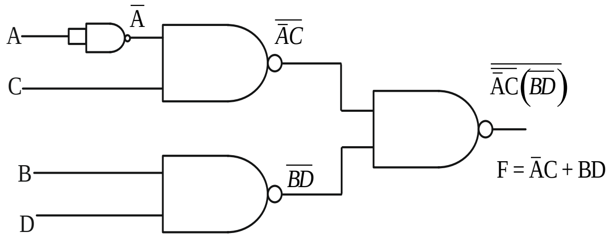
$f = \bar{A}BD + \bar{A}CD + \bar{A}C\bar{D} + AB\bar{C}D + ABCD$

$f = \bar{A}BCD + \bar{A}B\bar{C}D + \bar{A}BCD + \bar{A}\bar{B}CD + \bar{A}BC\bar{D}$
$+ \bar{A}\bar{B}C\bar{D} + AB\bar{C}D + ABCD$

k-map of 4 variables

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	1	1	1
11	0	1	1	0
10	0	0	0	0

$f = \bar{A}C + BD$



Hence, 4 NAND gate required.

4. (265)

Number of states in NFA = n + 1
= 8 +1 = 9
Number of states in minimal DFA = 2^8 = 256
Total states (A + B) = 256 + 9 = 265.

5. (82)

Since each region has at least five edges in its boundary, 2|E| > 5(53), or |E| ≥ (1/ 2)(5)(53), and from Theorem, we have
$|V| = |E| - 53 + 2 = |E| - 51 \geq (1/2)(5)(53) - 51 = (256/2) - 51 = 81\frac{1}{2}$. Hence |V| ≥ 82.

6. (b)

printf("%s", *(p+3)); //2025 is printed.
printf("%s", *p++); //Parakram is printed. Now p would points to 2024
printf("%s", *(p+++2));//2025 is printed. p then points to Shreshth
printf("%c", **p);//S is printed.

Output: 2025Parakram2025S

7. (a, b)

The given graph is a hamiltonian circuit, it covers all the vertices.
The chromatic number of the graph is 3.
The graph is not an eulerian as the degree of the vertices is not even.

8. (b, c, d)

$(0001 \quad 1101 \,.\, 1100)_2 = (1\,D.C)_{16}$

$\uparrow \quad \uparrow \quad \uparrow$

1 D C

Option (a) is not correct

$(001 \quad 001 \,.\, 100)_2 = (11.4)_8$

$\uparrow \quad \uparrow \quad \uparrow$

1 1 4

Option (b) is correct

$$(112)_4 = \left(1\times 4^2 + 1\times 1^1 + 2\times 4^0\right)_{10} = (22)_{10}$$

and $(211)_3 = \left(2\times 3^2 + 1\times 3^1 + 1\times 3^0\right)_{10} = (22)_{10}$

Option (c) is correct.

$$(214)_5 = \left(2\times 5^2 + 1\times 5^1 + 4\times 5^0\right)_{10} = (59)_{10}$$

and $(3B)_{16} = \left(3\times 16^1 + 11\times 16^0\right)_{10} = (59)_{10}$

Option (d) is correct.

9. (c)

- Set of real numbers are uncountable and not Turing acceptable.
- Set of prime numbers are decidable and acceptable by Turing machine.

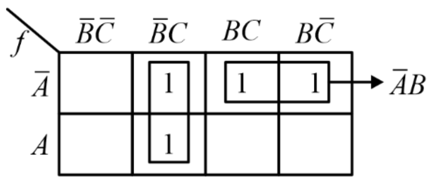
10. (d)

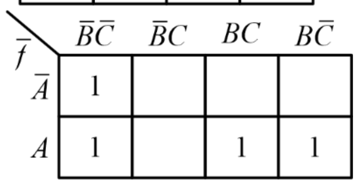
Given $f(A,B,C) = \overline{A}B + \overline{B}C$

$\therefore f_1(A,B,C) = f(f, \overline{f}, A)$

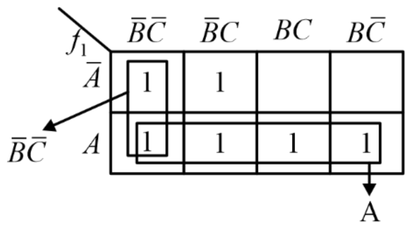
$f_1(A,B,C) = \overline{f}.\overline{f} + \overline{\overline{f}}.A = \overline{f} + fA$

$f_1(A,B,C) = \left(\overline{f} + f\right).\left(\overline{f} + A\right) = \left(\overline{f} + A\right)$





Therefore K-Map of $f_1 = \overline{f} + A$



$f_1(A,B,C) = A + \overline{B}\overline{C}$

11. (a, c, d)

Provided the degree sequence 5, p, 4, 4, 3, q, 2, 1 According to hand-shaking lemma, sum of degree = 2* Number of edges.

5 + p + 4 + 4 + 3 + q + 2 + 1 = 2*edges

19 + p + q = 2*13

19 + p + q = 26

p + q = 7.

Possible value of p and q: 7, 0
6, 1
5, 2
4, 3

Out of these 4 possibilities only 3 will make our degree sequence graphical i.e. {6,1}, {5,2} and {4, 3}.

12. (a)

(a) Halting problem is always undecidable.

(b) Emptiness problem in CFL is always decidable.

(c) Equivalence problem for DCFL is always decidable.

Hence, option (a) is correct.

13. (12)

fun1(5):

if(n < 2) return; //5 < 2 is FALSE

fun2(n-2); //fun2(3) is called.

printf("%d\t", n-2); //3 is printed./*3rd printf() executed*/

fun2(3):

if(n < 1) return;//3 < 1 FALSE

printf("%d\t", n-1);//2 is printed./*1st printf() executed*/

fun1(n – 1);//fun1(2) is called.

fun1(2):

if(n < 2) return; //2 < 2 is FALSE

fun2(n – 2); //fun2(0) is called. It simply returns.

printf("%d\t", n – 2); //0 is printed./*2nd printf() executed*/

fun2(5):

if(n < 1) return; //5 < 1 is FALSE

printf("%d\t", n – 1); //4 is printed./*4th printf() executed*/

fun1(n – 1);//fun1(4) is called.

fun1(4):

if(n < 2) return; //4 < 2 is FALSE

fun2(n – 2); //fun2(2) is called.

printf("%d\t", n – 2); //2 is printed./*6th printf() executed*/

fun2(2):

if(n < 1) return; //2 < 1 is FALSE

printf("%d\t", n – 1); //1 is printed./*5th printf() executed*/

fun1(n – 1);//fun1(1) is called. It simply returns.

Output: 2 0 3 4 1 2

Sum: 12

14. (2)

Given an initial state of 0110, the status of the counter will be 0110 after every 16, 32, 48, 64, and so forth clock pulses. The down counter will then do what the states indicate, as shown below.

After clock pulses	64	65	66	67	68
State of counter	0110	0101	0100	0011	0010
Decimal Value	6	5	4	3	2

15. (b)

$L(G_1)$ = $\{0^n1^n1^m\,2^m \mid m, n \geq 0\}$

= $\{0^n1^{m+n}\,2^m \mid m, n \geq 0\}$

$L(G_2)$ = $\{0^m1^n\,2^n1^m \mid m, n \geq 0\}$

$L(G_3)$ = $\{0^n\,1^{m+n}\,2^m \mid m, n \geq 0\}$

Hence, option (b) is correct.

16. (12)

fun1(5):

if(n < 2) return;// 5 < 2 is FALSE

fun2(n – 2); //fun2(3) is called.

printf("%d\t", n – 2); // 3 is printed./*3rd printf() executed*/

fun2(3):

if(n < 1) return; //3 < 1 FALSE

printf("%d\t", n – 1); // 2 is printed./*1st printf() executed*/

fun1(n – 1); //fun1(2) is called.

fun1(2):

if(n < 2) return;// 2 < 2 is FALSE

fun2(n – 2); // fun2(0) is called. It simply returns.

printf("%d\t", n – 2); //0 is printed./*2nd printf() executed*/

fun2(5):

if(n<1) return;// 5 < 1 is FALSE

printf("%d\t", n –1);//4 is printed./*4th printf() executed*/

fun1(n – 1); // fun1(4) is called.

fun1(4):

if(n < 2) return; // 4 < 2 is FALSE

fun2(n – 2); // fun2(2) is called.

printf("%d\t", n – 2); // 3 is printed./*6th printf() executed*/

fun2(2):

if(n < 1) return; // 2 < 1 is FALSE

printf("%d\t", n – 1); // 1 is printed./*5th printf() executed*/

fun1(n – 1); // fun1(1) is called. It simply returns.

Output: 2 0 3 4 1 2

Sum: 12

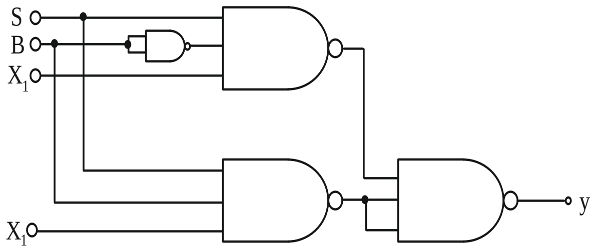
17. (c)

As per design specification, the expressions for output y can be obtained as

$$y = S\bar{B}x_1 + SBx_0$$

$$\bar{y} = \left(\overline{S\bar{B}\,x_1}\right)\left(\overline{SBx_0}\right) \text{ and } y = \overline{\left(\overline{S\bar{B}\,x_1}\right)\left(\overline{SBx_0}\right)}$$

The NAND gate implementation is show below.



18. (c)

void *p;

char *c="GATEWallah";

int a=513;// Binary of 513 is $(00000010\ 00000001)_2$

p = &a;

printf("%d ", *(char*)p); // Decimal equivalent of 0000001 is printed, i.e 1

p = c; //p stores the base address of the string GATEWallah

printf((char*)p + 4);// Wallah is printed.

return 0;

Output: 1Wallah

19. (a, b, c)

(a) $L_1 \oplus L_2 = (\overline{L_1} \cup L_2) \cup (L_1 \cup \overline{L_2})$

$= \left(\overline{\text{CFL}} \cup \text{DCFL}\right) \cup \left(\text{CFL} \cup \overline{\text{DCFL}}\right)$

= (CSL ∪ DCFL) ∪ (CFL ∪ CFL)

= CSL ∪ CFL

= CSL **Incorrect**

CSL but not CFL

(b) $\left(L_1 \cup \overline{L_3}\right) = \left(CFL \cup \overline{RE}\right)$

= (CFL ∪ Need Not RE)

= Not RE **Incorrect**

(c) $\left(L_2 \cup \overline{L_3}\right) = \left(DCFL \cup \overline{RE}\right)$

= DCFL ∪ Need Not RE

= Not RE **Incorrect**

(d) (L1 ∪ L2 ∪ L3) = CFL ∪ DCFL ∪ RE

= CFL ∪ CFL ∪ RE

= CFL ∪ RE

= RE **Correct**

Hence option (a, b, c) are correct.

20. (a)

```
void main()
{
    int i=printf("Parakram 2024");//i=13
    i=func(i=func(i=func(--i)));
    //func(--i) i.e func(12) returns 9.
    //func(i=9) i.e func(9) returns 6.
    //func(i=6) i.e func(6) returns 3.
    printf("%d", i);//3
}
```

Output: Parakram 20243

21. (312)

In BCD adder. Let the augend be $A = a_3a_2a_1a_0$ and addend be $B = b_3b_2b_1b_0$. Then A and B can vary from 00 to 99 for carry in $C_{in} = 0$ and 00 to 99 for $C_{in} = 1$. Total number of input combinations $= 2^9 = 512$ and total number of valid combinations = 200. So, don't care input combination = 512 – 200 = 312.

22. (b, d)

P = ab + aQ + bR

Q = baQ + bR

R = Raba + a

Apply Arden's Theorem:

$R = a(aba)^*$

$Q = (ba)^*bR$

$Q = (ba)^*ba(aba)^*$

P = ab + aQ + bR

P = aQ | bR | ab

= a[(ba)*ba(aba)*] + ba(aba)* + ab

$r * r = r^+$

$(ba)^*ba = (ba)^+$

$P = a(ba)+ (aba)^* + ba(aba)^* + ab$

Exactly match with option (d)

$P = a[(ba)^* \underline{ba(aba)^*}] + \underline{ba(aba)^*} + ab$

$P = [a(ba)^* + \in]ba(aba)^* + ab$

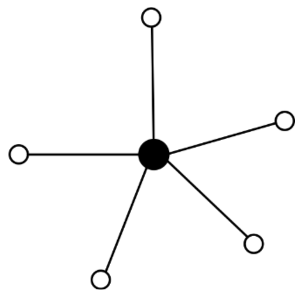
$= ab + [\in + a(ba)^*]ba(aba)^*$

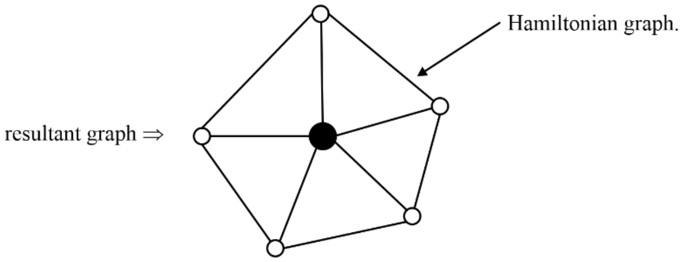
Exactly match with option (b).

Hence, option (b, d) are correct.

23. (c)

Given graph is $K_{1.n}$





24. (18)

```
func():
int x=-5;
static int y; // y=0
do
{
        x++; //x=-4, -2, 0, 2
        y++; //y=1, 2, 3, 4
}
while(x++<1);
    -4<1→ TRUE. x increments to -3
    -2<1→ TRUE, x increments to -1
    0<1 → TRUE, x increments to 1
    2<1 → FALSE. x increments to 3 STOP
printf("%d\t%d\t", x,y); //3 4 is printed.
func():
```

```
int x=-5;
static int y; // y=4
do
{
        x++; //x=-4, -2, 0, 2
        y++; //y=5, 6, 7, 8
}
while(x++<1);
```
-4<1→ TRUE. x increments to -3
-2<1 → TRUE, x increments to -1
0<1 → TRUE, x increments to 1
2<1 → FALSE. x increments to 3 STOP
printf("%d\t%d\t", x,y);// 3 8 is printed.

Output: 3 4 3 8

Sum: 18

25. **(4)**

$L_1 = aa^* b^+$
$= a^+ b^+$
$L_2 = ab\,(ab)^*$
$= (ab)^+$

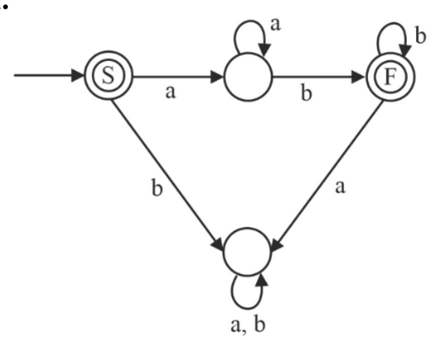
$L_2^* = \left[(ab)^+\right]^*$

$L_2^* = (ab)^*$

$L_3 = \dfrac{a^+b^+}{(ab)^*}$

$= a^+ b^+ + \in$
$= aa^* bb^* + \in$

DFA:



Number of states = 4

26. **(b, c, d)**

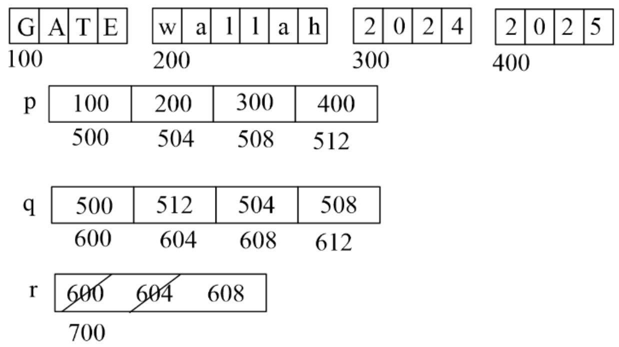
The counter will have the number of distinct states equal to 256, mod = 256.

The frequency of output of eighth FF will be 512/256 = 2 kHz.

Starting from 0000 0000 the counter will be back to 0000 0000 every after 256 clock pulses.

After 512 clock pulses, the state will be 0000 0000 and after 520 (512 + 8) clock pulses, the state will be 0000 1000.

27. **(d)**



printf (“%S”, **++r);
**604
*512
*400 → 2025 is printed.

printf (“%S”, *_ _*++r+2)
*_ _*608 + 2
*_ _ 504 + 2
*500 + 2
102 → TE is printed.

printf (“%S”, *p[-1] + 3)
**604 + 3
*512 + 3 = 403 → 5 is printed.

printf(“%S”, p[-1][-1]+1); //301 → 024 is printed.
Output: 2025 TE 5 024

28. **(1)**

- If the language is CFL, DCFL, CSL and recursive then language may /may not be regular. So, the problem is **undecidable**.
- If the language is regular then surely language will be DCFL, CFL, CSL recursive and RE also. Hence, the problem is **decidable**.
- NPDA $\geq$ DPDA

  For every DPDA, NPDA exits but vice versa not true. **(Decidable)**.
- Type-3 is regular grammar. **(Decidable)**.

29. **(a)**

f():
static int a=3;
int b=5;

a ≤ b++; //a = 3-5=-2; b is incremented to 6.

printf("%d\t%d\n",a,b); //-2 6 is printed.

f():
static int a=3; // a contains -2.

int b=5;

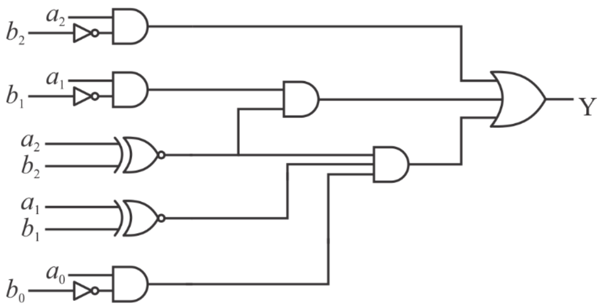
a-=b++; //a=-2-5=-7; b is incremented to 6.

printf("%d\t%d\n",a,b); //-7 6 is printed.
main():
static int a=2;
int b=1;
a+=3; //a=5
printf("%d\t%d\n",a,b); //5 1 is printed.

$$\text{Output:}\ \begin{matrix} -2 & 6 \\ -7 & 6 \\ 5 & 1 \end{matrix}$$

30. (b)

Lets draw the circuit and analyze it :



Output Y is

$$Y = a_2\overline{b}_2 + (a_2\ b_2)a_1 \odot \overline{b}_1 + (a_2 \odot b_2)(a_1 \odot b_1)a_0\overline{b}_0$$

- This output is nothing but a comparator output when designed for (A > B) so output will be ‘1’ when (A > B).
- For $A < B \rightarrow$

$$y_1(A < B) = \overline{a}_2 b_2 + (a_2 \odot b_2)\overline{a}_1 b_1 + (a_2 \odot b_2)(a_1 \odot b_1)\overline{a}_0 b_0$$

- For $A = B \rightarrow$

$$y_1(A = B) = (a_2 \odot b_2)(a_1 \odot b_1)(a_0 \odot b_0)$$

Hence, output Y will be ‘1’ when A > B.