# Programming and Data structures

1.

Hindi [CLICK HERE]
English [CLICK HERE]

2.

Hindi [CLICK HERE]
English [CLICK HERE]

3.

Hindi [CLICK HERE]
English [CLICK HERE]

4.

Hindi [CLICK HERE]
English [CLICK HERE]

5.

Hindi [CLICK HERE]
English [CLICK HERE]

6.

Hindi [CLICK HERE]
English [CLICK HERE]

7.

Hindi [CLICK HERE]

English [CLICK HERE]

8.

Hindi [CLICK HERE]
English [CLICK HERE]

9. Consider the following C function definition

```
int Trial (int  a, int  b, int  c)
{
  if ((a>=b) && (c<b)) return  b;
  else if (a>=b) return Trial(a,  c,  b);
  else return Trial(b,  a,  c);
}
```

The functional Trial:

**(A)** finds the maximum of a, b and c
**(B)** finds the minimum of a, b and c
**(C)** finds the middle number of a, b and c
**(D)** None of the above

Hindi [CLICK HERE]
English [CLICK HERE]

10. SAME AS QUESTION 1.1 IN CAO

The most appropriate matching for the following pairs is:-

| | |
|---|---|
| X: Indirect addressing | 1: Loops |
| Y: Immediate addressing | 2: Pointers |
| Z: Auto decrement addressing | 3: Constants |

a)X−3,Y−2,Z−1

b)X−1,Y−3,Z−2
c)X−2,Y−3,Z−1
d)X−3,Y−1,Z−2

Hindi [CLICK HERE]
English [CLICK HERE]

11.

Hindi [CLICK HERE]
English [CLICK HERE]

12.

Hindi [CLICK HERE]
English [CLICK HERE]

13.

Hindi [CLICK HERE]
English [CLICK HERE]

14.

Hindi [CLICK HERE]
English [CLICK HERE]

15. The value of j at the end of the execution of the following C program:

```
int incr (int i)
{
   static int count = 0;
```

```
    count = count + i;
    return (count);
}
main ()
{
    int i, j;
    for (i = 0; i <= 4; i++)
       j = incr (i);
}
```
is:

(A)10
(B)4
(C)6
(D)7
Hindi [CLICK HERE]
English [CLICK HERE]

16. What is printed by the print statements in the program P1 assuming call by reference parameter passing?

```
Program P1()
{
   x = 10;
   y = 3;
   func1(y,x,x);
   print x;
   print y;
}
func1(x,y,z)
{
   y = y+4;
   z = x+y+z;
}
```
(A) 10, 3
(B) 31, 3
(C) 27, 7
(D) None of the above

Hindi [CLICK HERE]
English [CLICK HERE]

17.

Hindi [CLICK HERE]
English [CLICK HERE]

18. Consider the following program

```
Program P2
  var n : int;

    procedure W(var x : int)
  begin
          x = x +1;
    print x;
  end

    procedure D
  begin
    var n : int;
          n = 3;
          W(n);
  end

  begin     \\begin P2
          n=10;
          D;
  end
```

If the language has dynamic scooping and parameters are passed by reference, what will be printed by the program?

1. 10
2. 11
3. 3
4. None of the above

Hindi [CLICK HERE]
English [CLICK HERE]

19.

Hindi [CLICK HERE]
English [CLICK HERE]

20.

Hindi [CLICK HERE]
English [CLICK HERE]

21.

Hindi [CLICK HERE]
English [CLICK HERE]

22.

Hindi [CLICK HERE]
English [CLICK HERE]

23.

Hindi [CLICK HERE]
English [CLICK HERE]

24. Consider the C program shown below:

```
#include<stdio.h>
#define print(x) printf("%d", x)

int x;
void Q(int z)
{
        z+=x;
    print(z);
}
```

```
void P(int *y)
{
    int x = *y + 2;
        Q(x);
    *y = x - 1;
    print(x);
}
main(void) {
        x = 5;
        P(&x);
    print(x);
}
```

The output of this program is

(A) 12 7 6
(B) 22 12 11
(C) 14 6 6
(D) 7 6 6

Hindi [CLICK HERE]
English [CLICK HERE]

25.

Hindi [CLICK HERE]
English [CLICK HERE]

26.

Hindi [CLICK HERE]
English [CLICK HERE]

27.

Hindi [CLICK HERE]

English [CLICK HERE]

28.

Hindi [CLICK HERE]
English [CLICK HERE]

29.

Hindi [CLICK HERE]
English [CLICK HERE]

30.

Hindi [CLICK HERE]
English [CLICK HERE]

31.

Hindi [CLICK HERE]
English [CLICK HERE]

32.

Hindi [CLICK HERE]
English [CLICK HERE]

33.

Hindi [CLICK HERE]
English [CLICK HERE]

34.
Consider the following C program segment:

```
char p[20];
char *s = "string";
int length = strlen(s);
int i;
for (i = 0; i < length; i++)
     p[i] = s[length — i];
printf("%s",p);
```

The output of the program is

(A) gnirts
(B) gnirt
(C) string
(D) no output is printed

Hindi [CLICK HERE]
English [CLICK HERE]

35.
Hindi [CLICK HERE]
English [CLICK HERE]

36.
Hindi [CLICK HERE]
English [CLICK HERE]

37.
Hindi [CLICK HERE]
English [CLICK HERE]

38. Consider the following C-program:

```
void foo(int n, int sum)
{
  int k = 0, j = 0;
  if (n == 0) return;
  k = n % 10;
  j = n / 10;
  sum = sum + k;
  foo (j, sum);
  printf ("%d,", k);
}

int main ()
{
  int a = 2048, sum = 0;
  foo (a, sum);
  printf ("%d\n", sum);

}
```

What does the above program print?
**(A)** 8, 4, 0, 2, 14
**(B)** 8, 4, 0, 2, 0
**(C)** 2, 0, 4, 8, 14
**(D)** 2, 0, 4, 8, 0

Hindi [CLICK HERE]
English [CLICK HERE]

39.
Hindi [CLICK HERE]
English [CLICK HERE]

40.
Hindi [CLICK HERE]
English [CLICK HERE]

41.
Hindi [CLICK HERE]
English [CLICK HERE]

42.
   Hindi [CLICK HERE]
   English [CLICK HERE]

43.
   Hindi [CLICK HERE]
   English [CLICK HERE]

44.
   Hindi [CLICK HERE]
   English [CLICK HERE]

45.
   Hindi [CLICK HERE]
   English [CLICK HERE]

46.
   Hindi [CLICK HERE]
   English [CLICK HERE]

47.
   Hindi [CLICK HERE]
   English [CLICK HERE]

48.
   Hindi [CLICK HERE]
   English [CLICK HERE]

49.
   Hindi [CLICK HERE]
   English [CLICK HERE]

50.
   Hindi [CLICK HERE]
   English [CLICK HERE]

51.
   Hindi [CLICK HERE]
   English [CLICK HERE]

52. 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?

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

**(A)** 9

**(B)** 8

**(C)** 7

**(D)** 6

Hindi [CLICK HERE]
English [CLICK HERE]

53.
Hindi [CLICK HERE]
English [CLICK HERE]

54.
Hindi [CLICK HERE]
English [CLICK HERE]

55.
Hindi [CLICK HERE]
English [CLICK HERE]

56.
Hindi [CLICK HERE]
English [CLICK HERE]

57. What is the output printed by the following C code?

```
# include  stdio.h
int main ()
{
   char a [6] = "world";
   int i, j;
   for (i = 0, j = 5; i < j; a [i++] = a [j--]);
   printf ("%s\n", a);
}
```

(A) dlrow
(B) Null String
(C) dlrld
(D) worow

Hindi [CLICK HERE]
English [CLICK HERE]

58.
Hindi [CLICK HERE]
English [CLICK HERE]

59.
indi [CLICK HERE]
English [CLICK HERE]

60.
Hindi [CLICK HERE]
English [CLICK HERE]

61.
Hindi [CLICK HERE]
English [CLICK HERE]

62.
Hindi [CLICK HERE]
English [CLICK HERE]

63.
Hindi [CLICK HERE]
English [CLICK HERE]

64.
Hindi [CLICK HERE]
English [CLICK HERE]

65.
Hindi [CLICK HERE]
English [CLICK HERE]

66.
Hindi [CLICK HERE]
English [CLICK HERE]

67.
Hindi [CLICK HERE]
English [CLICK HERE]

68.
Hindi [CLICK HERE]
English [CLICK HERE]

69.
Hindi [CLICK HERE]
English [CLICK HERE]

70.
Hindi [CLICK HERE]
English [CLICK HERE]

71.
Hindi [CLICK HERE]
English [CLICK HERE]

72.
Hindi [CLICK HERE]
English [CLICK HERE]

73.
Hindi [CLICK HERE]
English [CLICK HERE]

74.
Hindi [CLICK HERE]
English [CLICK HERE]

75.

Hindi [CLICK HERE]
English [CLICK HERE]

76.

Hindi [CLICK HERE]
English [CLICK HERE]

77.

Hindi [CLICK HERE]
English [CLICK HERE]

78.

Hindi [CLICK HERE]
English [CLICK HERE]

79.

Hindi [CLICK HERE]
English [CLICK HERE]

80.

Hindi [CLICK HERE]
English [CLICK HERE]

81.

Hindi [CLICK HERE]
English [CLICK HERE]

82.

Hindi [CLICK HERE]
English [CLICK HERE]

83.

Hindi [CLICK HERE]
English [CLICK HERE]

84.

Hindi [CLICK HERE]
English [CLICK HERE]

85.

Hindi [CLICK HERE]

English [CLICK HERE]

86.
Hindi [CLICK HERE]
English [CLICK HERE]

87.
Hindi [CLICK HERE]
English [CLICK HERE]

88.
Hindi [CLICK HERE]
English [CLICK HERE]

89.
Hindi [CLICK HERE]
English [CLICK HERE]

90.
Hindi [CLICK HERE]
English [CLICK HERE]

91.
Hindi [CLICK HERE]
English [CLICK HERE]

92.
Hindi [CLICK HERE]
English [CLICK HERE]

93.
Hindi [CLICK HERE]
English [CLICK HERE]

94.
Hindi [CLICK HERE]
English [CLICK HERE]

95.
Hindi [CLICK HERE]
English [CLICK HERE]

96.

Hindi [CLICK HERE]
English [CLICK HERE]

97.
Hindi [CLICK HERE]
English [CLICK HERE]

98.
Hindi [CLICK HERE]
English [CLICK HERE]

99.
Hindi [CLICK HERE]
English [CLICK HERE]

100.
Hindi [CLICK HERE]
English [CLICK HERE]

101.
Hindi [CLICK HERE]
English [CLICK HERE]

102.
Hindi [CLICK HERE]
English [CLICK HERE]

103. Consider the following two functions

```
void fun1(int n) {
  if(n == 0) return;
    printf("%d", n);
    fun2(n - 2);
    printf("%d", n);
}
void fun2(int n) {
  if(n == 0) return;
    printf("%d", n);
    fun1(++n);
    printf("%d", n);
}
```

The output printed when fun1 (5) is called is

(A) 53423122233445

(B) 53423120112233

(C) 53423122132435

(D) 53423120213243

Hindi [CLICK HERE]
English [CLICK HERE]

104. Match the following:

| list 1 | list 2 |
|---|---|
| (P) static char var; | (i) Sequence of memory locations to store addresses |
| (Q) m = malloc (10); m = NULL; | (ii) A variable located in data section of memory |
| (R) char *ptr[10]; | (iii) Request to allocate a cpu register to store data |
| (S) register int var1; | (iv) a lost memory which cannot be freed |

P-ii; Q-iv; R-i; S-iii
P-ii; Q-i; R-iv; S-iii
P-ii; Q-iv; R-iii; S-i
P-iii; Q-iv; R-i; S-ii

Hindi [CLICK HERE]
English [CLICK HERE]

105. Consider the following function implemented in C:

```
void printxy(int x, int y) {
  int *ptr;
    x=0;
    ptr=&x;
    y=*ptr;
  *ptr=1;
    printf(“%d, %d”, x, y);
}
```

The output of the printxy(1,1) is

**(A)** 0,0
**(B)** 0,1
**(C)** 1,0
**(D)** 1,1

Hindi [CLICK HERE]
English [CLICK HERE]

106. Consider the following program

```
#include<stdio.h>
#include<string.h>

int main()
{
    char * c = "GATECSIT2017";
    char *p = c;
    printf("%d", (int)strlen(c+2[p]-6[p]-1));
    return 0;
}
```

The Output of the following program is____

Hindi [CLICK HERE]
English [CLICK HERE]

107. Consider the following C program:

```
#include
int main()
{
    int m = 10;
    int n, n1;
    n = ++m;
    n1 = m++;
    n--;
    --n1;
```

```
        n -= n1;
        printf("%d",n);
        return 0;
}
```

The output of the program is ______.

Hindi [CLICK HERE]
English [CLICK HERE]

108.
Hindi [CLICK HERE]
English [CLICK HERE]

109. Consider the following C code:

```
#include<stdio.h>
int *assignval (int *x, int val) {
  *x = val;
  return x;
}

void main () {
  int *x = malloc(sizeof(int));
  if (NULL == x) return;
    x = assignval (x,0);
  if (x) {
          x = (int *)malloc(sizeof(int));
    if (NULL == x) return;
          x = assignval (x,10);
  }
    printf("%d\n", *x);
    free(x);
}
```

The code suffers from which one of the following problems:
**(A)** compiler error as the return of malloc is not typecast appropriately.
**(B)** compiler error because the comparison should be made as x==NULL and not as shown.
**(C)** compiles successfully but execution may result in dangling pointer.

**(D)** compiles successfully but execution may result in memory leak.

Hindi [CLICK HERE]
English [CLICK HERE]

110. Consider the C program fragment below which is meant to divide x by y using repeated subtractions. The variable x, y, q and r are all unsigned int.

```
while (r >= y) {
    r=r-y;
    q=q+1;
}
```

Which of the following conditions on the variables x, y, q and r before the execution of the fragment will ensure that the loop terminates in a state satisfying the condition x == (y*q + r)?

(A) ( q == r ) && ( r == 0)
(B) ( x > 0 ) && ( r == x ) && ( y > 0 )
(C) ( q == 0 ) && ( r == x ) && ( y > 0 )
(D) ( q == 0 ) && ( y > 0 )

Hindi [CLICK HERE]
English [CLICK HERE]

111. Consider the following C program.

```
#include<stdio.h>
struct Ournode{
  char x, y, z;
};
int main() {
  struct Ournode p={'1', '0', 'a'+2};
  struct Ournode *q=&p;
    printf("%c, %c", *((char*)q+1), *((char*)q+2));
  return 0;
}
```

The output of this program is:
**(A)** 0, c
**(B)** 0, a+2
**(C)** '0', 'a+2'
**(D)** '0', 'c'

Hindi [CLICK HERE]
English [CLICK HERE]

112. Consider the following C program:

```
#include<stdio.h>

int counter=0;

int calc (int a, int b) {
   int c;
        counter++;
   if(b==3) return (a*a*a);
   else {
                c = calc(a, b/3);
      return (c*c*c);
   }
}

int main() {
        calc(4, 81);
        printf("%d", counter);
}
```

The output of this program is ________ .

Hindi [CLICK HERE]
English [CLICK HERE]

113. Consider the following program written in pseudo-code. Assume that

x and y are integers.

```
Count (x, y) {
  if (y !=1 ) {
    if (x !=1) {
      print("*");
      Count (x/2, y);
    }
    else {
          y=y-1;
      Count (1024, y);
    }
  }
}
```

The number of times that the print statement is executed by the call Count(1024, 1024) is ________ .

Hindi [CLICK HERE]
English [CLICK HERE]

114. Consider the following C program:

```
#include stdio.h
void fun1(char *s1, char *s2) {
  char *temp;
  temp = s1;
  s1 = s2;
  s2 = temp;
}
void fun2(char **s1, char **s2) {
  char *temp;
  temp = *s1;
  *s1 = *s2;
  *s2 = temp;
}
int main() {
  char *str1 = "Hi", *str2 = "Bye";
  fun1(str1, str2);
  printf("%s %s", str1, str2);
  fun2(&str1, &str2);
  printf("%s %s", str1, str2);
  return 0;
}
```

The output of the program above is

(A) Hi Bye Bye Hi
(B) Hi Bye Hi Bye
(C) Bye Hi Hi Bye
(D) Bye Hi Bye Hi

Hindi [CLICK HERE]
English [CLICK HERE]

115. Consider the following C code. Assume that unsigned long int type length is 64 bits

```
unsigned long int fun(unsigned long int n)
{
    unsigned long int i, j=0, sum = 0;
    for( i=n; i>1; i=i/2) j++;
    for( ; j>1; j=j/2) sum++;
    return sum;
}
```

The value returned when we call fun with the input 2^40 is

**(A)** 4
**(B)** 5
**(C)** 6
**(D)** 40

Hindi [CLICK HERE]
English [CLICK HERE]

116. Consider the following C program:

```
#include stdio.h
 int main(){
  int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 5}, *ip = arr + 4;

  printf("%d\n", ip[1]);

  return 0;
```

}
The number that will be displayed on execution of the program is ________ .

Hindi [CLICK HERE]
English [CLICK HERE]

117. Consider the following C program :

```
#include<stdio.h>
int jumble(int x, int y){
    x = 2*x+y;
  return x;
}
int main(){
  int x=2, y=5;
    y=jumble(y,x);
    x=jumble(y,x);
    printf("%d \n",x);
  return 0;
}
```

The value printed by the program is ____________.

Hindi [CLICK HERE]
English [CLICK HERE]

118. Consider the following C function.

```
void convert (int n )
{
    if (n<0)
                printf{“%d”, n);
    else
        {
                convert(n/2);
                printf(“%d”, n%2);
    }
}
```

Which one of the following will happen when the function convert is called with any positive integer n as argument?

**(A)** It will print the binary representation of n in the reverse order and terminate.

**(B)** It will print the binary representation of n but will not terminate

**(C)** It will not print anything and will not terminate.

**(D)** It will print the binary representation of n and terminate.

Hindi [CLICK HERE]
English [CLICK HERE]

119. Consider the following C program:

```c
#include <stdio.h>
int main() {
  float sum = 0.0, j=1.0, i=2.0;
  while (i/j > 0.0625) {
        j=j+j;
        sum=sum+i/j;
        printf("%f\n", sum);
  }
  return 0;
}
```

The number of times the variable sum will be printed, when the above program is executed, is ________

Hindi [CLICK HERE]
English [CLICK HERE]

120. Consider the following C program:

```c
#include <stdio.h>
int r()
{
   static int num=7;
   return num--;
}
int main()
{
   for (r();r();r())
              printf(“%d”,r());
   return 0;
}
```

Which one of the following values will be displayed on execution of the programs?

1. 41
2. 52
3. 63
4. 630

Hindi [CLICK HERE]
English [CLICK HERE]

121. Consider the following C program:

- `#include <stdio.h>`
- `int main()`
- `{`
- `  int a[] = {2, 4, 6, 8, 10};`
- `  int i, sum=0, *b=a+4;`
- `  for (i=0; i<5; i++)`
- `          sum=sum+(*b-i)-*(b-i);`
- `    printf("%d\n", sum);`
- `  return 0;`
- `}`

The output of the above C program is ______

Hindi [CLICK HERE]
English [CLICK HERE]

122. Consider the following C functions.

```
int tob (int b, int* arr)
{
    int i;
    for (i = 0; b>0; i++)
    {
        if (b%2)  arr [i] = 1;
        else          arr[i] = 0;
        b = b/2;
    }
    return (i);
}


int pp(int a, int b)
{
    int  arr[20];
    int i, tot = 1, ex, len;
    ex = a;
    len = tob(b, arr);
```

```
        for (i=0; i<len ; i++)
        {
             if (arr[i] ==1)
                  tot = tot * ex;
             ex= ex*ex;
        }
        return (tot) ;
    }
```

The value returned by pp(3,4) is ________ .
**Note –** This question was Numerical Type.
**(A)** 81
**(B)** 64
**(C)** 100
**(D)** 49

Hindi [CLICK HERE]
English [CLICK HERE]

123.

Consider the following C functions.

```
int fun1(int n)
{
    static int i= 0;
    if (n > 0)
    {
       ++i;
       fun1(n-1);
    }
  return (i);
}


int fun2(int n)
{
   static int i= 0;
   if (n>0)
```

```
    {
       i = i+ fun1 (n) ;
       fun2(n-1) ;
    }
 return (i);
 }
```

The return value of fun2(5) is _________ .

**Note –** This question was Numerical Type.

**(A)** 55

**(B)** 45

**(C)** 50

**(D)** 52

Hindi [CLICK HERE]

English [CLICK HERE]

1.
Hindi [CLICK HERE]
English [CLICK HERE]

2.
Hindi [CLICK HERE]
English [CLICK HERE]

3.
Hindi [CLICK HERE]
English [CLICK HERE]

4.
Hindi [CLICK HERE]
English [CLICK HERE]

5.
Hindi [CLICK HERE]
English [CLICK HERE]

6.
Hindi [CLICK HERE]
English [CLICK HERE]

7.
Hindi [CLICK HERE]
English [CLICK HERE]

8.
Hindi [CLICK HERE]
English [CLICK HERE]

9.
Hindi [CLICK HERE]
English [CLICK HERE]

10.
Hindi [CLICK HERE]
English [CLICK HERE]

11.

Hindi [CLICK HERE]
English [CLICK HERE]

1.

   Hindi [CLICK HERE]
   English [CLICK HERE]

2.

   Hindi [CLICK HERE]
   English [CLICK HERE]

3.

   Hindi [CLICK HERE]
   English [CLICK HERE]

4.

   Hindi [CLICK HERE]
   English [CLICK HERE]

5.

   Hindi [CLICK HERE]
   English [CLICK HERE]

6.
   Hindi [CLICK HERE]
   English [CLICK HERE]

7.
   Hindi [CLICK HERE]
   English [CLICK HERE]

8.
   Hindi [CLICK HERE]
   English [CLICK HERE]

9.
   Hindi [CLICK HERE]
   English [CLICK HERE]

10.
   Hindi [CLICK HERE]
   English [CLICK HERE]

11.
   Hindi [CLICK HERE]
   English [CLICK HERE]

12.
   Hindi [CLICK HERE]
   English [CLICK HERE]

13.
   Hindi [CLICK HERE]
   English [CLICK HERE]

14.
   Hindi [CLICK HERE]
   English [CLICK HERE]

15.
   Hindi [CLICK HERE]
   English [CLICK HERE]

16.
   Hindi [CLICK HERE]

English [CLICK HERE]

17.
Hindi [CLICK HERE]
English [CLICK HERE]

18.
Hindi [CLICK HERE]
English [CLICK HERE]

19.
Hindi [CLICK HERE]
English [CLICK HERE]

20.
Hindi [CLICK HERE]
English [CLICK HERE]

21.
Hindi [CLICK HERE]
English [CLICK HERE]

22.
Hindi [CLICK HERE]
English [CLICK HERE]

23.
Hindi [CLICK HERE]
English [CLICK HERE]

24.
Hindi [CLICK HERE]
English [CLICK HERE]

25.
Hindi [CLICK HERE]
English [CLICK HERE]

26.
Hindi [CLICK HERE]
English [CLICK HERE]

27.

Hindi [CLICK HERE]
English [CLICK HERE]

28.
Hindi [CLICK HERE]
English [CLICK HERE]

1. h

   Hindi [CLICK HERE]
   English [CLICK HERE]

2.
   Hindi [CLICK HERE]
   English [CLICK HERE]

3.

Hindi [CLICK HERE]
English [CLICK HERE]

4.

Hindi [CLICK HERE]
English [CLICK HERE]

5.

Hindi [CLICK HERE]
English [CLICK HERE]

6.

Hindi [CLICK HERE]
English [CLICK HERE]

7.

Hindi [CLICK HERE]
English [CLICK HERE]

8.

Hindi [CLICK HERE]
English [CLICK HERE]

9. Hindi [CLICK HERE]
10. English [CLICK HERE]

1.

Hindi [CLICK HERE]
English [CLICK HERE]

2.

Hindi [CLICK HERE]
English [CLICK HERE]

3.

Hindi [CLICK HERE]

English [CLICK HERE]

4.
Hindi [CLICK HERE]
English [CLICK HERE]

5.
Hindi [CLICK HERE]
English [CLICK HERE]

6.
Hindi [CLICK HERE]
English [CLICK HERE]

7.
Hindi [CLICK HERE]
English [CLICK HERE]

8.
Hindi [CLICK HERE]
English [CLICK HERE]

9.
Hindi [CLICK HERE]
English [CLICK HERE]

10.

Hindi [CLICK HERE]
English [CLICK HERE]

11.
Hindi [CLICK HERE]
English [CLICK HERE]

12.
Hindi [CLICK HERE]
English [CLICK HERE]

13.
Hindi [CLICK HERE]
English [CLICK HERE]

14.

Hindi [CLICK HERE]
English [CLICK HERE]

15.

Hindi [CLICK HERE]
English [CLICK HERE]

16.

Hindi [CLICK HERE]
English [CLICK HERE]

17.

Hindi [CLICK HERE]
English [CLICK HERE]

18.

Hindi [CLICK HERE]
English [CLICK HERE]

19.

Hindi [CLICK HERE]
English [CLICK HERE]

20.

Hindi [CLICK HERE]
English [CLICK HERE]

21.

Hindi [CLICK HERE]
English [CLICK HERE]

22.

Hindi [CLICK HERE]
English [CLICK HERE]

23.

Hindi [CLICK HERE]
English [CLICK HERE]

24.

Hindi [CLICK HERE]
English [CLICK HERE]

25.
Hindi [CLICK HERE]
English [CLICK HERE]

26.
Hindi [CLICK HERE]
English [CLICK HERE]

27.
Hindi [CLICK HERE]
English [CLICK HERE]

28.

Hindi [CLICK HERE]
English [CLICK HERE]

29.
Hindi [CLICK HERE]
English [CLICK HERE]

30.
Hindi [CLICK HERE]
English [CLICK HERE]

31.
Hindi [CLICK HERE]
English [CLICK HERE]

32.
Hindi [CLICK HERE]
English [CLICK HERE]

33.
Hindi [CLICK HERE]
English [CLICK HERE]

34.
Hindi [CLICK HERE]
English [CLICK HERE]

35.
Hindi [CLICK HERE]

English [CLICK HERE]

36.
Hindi [CLICK HERE]
English [CLICK HERE]
37.

Hindi [CLICK HERE]
English [CLICK HERE]

38.
Hindi [CLICK HERE]
English [CLICK HERE]

39.
Hindi [CLICK HERE]
English [CLICK HERE]

40.
Hindi [CLICK HERE]
English [CLICK HERE]

41.
Hindi [CLICK HERE]
English [CLICK HERE]

42.
Hindi [CLICK HERE]
English [CLICK HERE]

43.
Hindi [CLICK HERE]
English [CLICK HERE]

44.
Hindi [CLICK HERE]
English [CLICK HERE]

45.
Hindi [CLICK HERE]
English [CLICK HERE]
46.

Hindi [CLICK HERE]
English [CLICK HERE]

47.
Hindi [CLICK HERE]
English [CLICK HERE]

48.
Hindi [CLICK HERE]
English [CLICK HERE]

49.
Hindi [CLICK HERE]
English [CLICK HERE]

50.
Hindi [CLICK HERE]
English [CLICK HERE]

51.
Hindi [CLICK HERE]
English [CLICK HERE]

52.
Hindi [CLICK HERE]
English [CLICK HERE]

53.
Hindi [CLICK HERE]
English [CLICK HERE]

54.
Hindi [CLICK HERE]
English [CLICK HERE]
55.

Hindi [CLICK HERE]
English [CLICK HERE]

56.
Hindi [CLICK HERE]
English [CLICK HERE]

57.

Hindi [CLICK HERE]
English [CLICK HERE]

58.

Hindi [CLICK HERE]
English [CLICK HERE]

59.

Hindi [CLICK HERE]
English [CLICK HERE]

60.

Hindi [CLICK HERE]
English [CLICK HERE]

1.

Hindi [CLICK HERE]
English [CLICK HERE]

2.

Hindi [CLICK HERE]
English [CLICK HERE]

3.

Hindi [CLICK HERE]
English [CLICK HERE]

4.

Hindi [CLICK HERE]
English [CLICK HERE]

5.

Hindi [CLICK HERE]
English [CLICK HERE]

6.

Hindi [CLICK HERE]
English [CLICK HERE]

7.
Hindi [CLICK HERE]
English [CLICK HERE]

8.
Hindi [CLICK HERE]
English [CLICK HERE]

9.
Hindi [CLICK HERE]
English [CLICK HERE]

10.

Hindi [CLICK HERE]
English [CLICK HERE]

11.
Hindi [CLICK HERE]
English [CLICK HERE]

12.
Hindi [CLICK HERE]
English [CLICK HERE]

13.
Hindi [CLICK HERE]
English [CLICK HERE]

14.
Hindi [CLICK HERE]
English [CLICK HERE]

15.
Hindi [CLICK HERE]
English [CLICK HERE]

16.
Hindi [CLICK HERE]
English [CLICK HERE]

17.
Hindi [CLICK HERE]

English [CLICK HERE]

18.
Hindi [CLICK HERE]
English [CLICK HERE]
19.

Hindi [CLICK HERE]
English [CLICK HERE]

20.
Hindi [CLICK HERE]
English [CLICK HERE]

21.
Hindi [CLICK HERE]
English [CLICK HERE]

22.
Hindi [CLICK HERE]
English [CLICK HERE]

23.
Hindi [CLICK HERE]
English [CLICK HERE]

1.

Hindi [CLICK HERE]
English [CLICK HERE]

2.
Hindi [CLICK HERE]
English [CLICK HERE]

3.
Hindi [CLICK HERE]

English [CLICK HERE]

4.
Hindi [CLICK HERE]
English [CLICK HERE]

5.
Hindi [CLICK HERE]
English [CLICK HERE]

6.
Hindi [CLICK HERE]
English [CLICK HERE]

7.
Hindi [CLICK HERE]
English [CLICK HERE]

8.
Hindi [CLICK HERE]
English [CLICK HERE]

9.
Hindi [CLICK HERE]
English [CLICK HERE]

10.

Hindi [CLICK HERE]
English [CLICK HERE]

11.
Hindi [CLICK HERE]
English [CLICK HERE]

12.
Hindi [CLICK HERE]
English [CLICK HERE]

13.
Hindi [CLICK HERE]
English [CLICK HERE]

14.
Hindi [CLICK HERE]
English [CLICK HERE]

15.
Hindi [CLICK HERE]
English [CLICK HERE]

16.
Hindi [CLICK HERE]
English [CLICK HERE]

of