Q.11 Consider the following function:

```
int unknown(int n)
{
        int i, j, k=0;
        for (i=n/2; i<=n; i++)
            for (j=2; j<=n; j=j*2)
                    k = k + n/2;
    return (k);
}
```

The return value of the function is

(A) (n^2) (B) $(n^2 \text{Logn})$ (C) (n^3) (D) $(n^3 \text{Logn})$

Q.12 What will be output when you will execute following c code?

```
#include<stdio.h>
void main()
{
    int a=3;
    if(a--,--a,a--,a)
                printf("Pankaj");
    else
                printf("Sharma");
}
```

(A) Pankaj (B) Sharma (C) Run time error (D) Compilation error

Q.13 What is the output of the following program?

```
#include<stdio.h>
int funcf (int x);
int funcg (int y);
main ()
  {
    int x=5, y=10, count;
    for (count = 1; count <=2; ++count)
```

```
        {
                y += funcf (x) + funcg (x);
                 printf ("%d", y) ;
        }
    }
     funcf (int x)
     {
     int y;
     y = funcg(g);
     return (y);
      }
     funcg (int x)
    {
     static int y = 10;
     y + 1= 1;
     return (y + x);
    }
```

(A) 43 80 (B) 42 74 (C) 33 37 (D) 32 32

Q.14 Consider the following C program

```
int a, b, c = 0;
void prtFun (void);
int main ()
{
    static int a = 1;          /* line 1 */
    prtFun();
    a += 1;
    prtFun();
    printf ("\n %d %d " , a, b) ;
}
void prtFun (void)
{
    static int a = 2;          /* line 2 */
    int b = 1;
    a += ++b;
    printf (" \n %d %d " , a, b);
}
```

What output will be generated by the given code segment?

(A) 3 1
4 1
4 2

(B) 4 2
6 1
6 1

(C) 4 2
6 2
2 0

(D) 3 1
5 2
5 2

Q.15 Consider the following C program:

```
#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?

(A) 41 (B) 52 (C) 63 (D) 630

Q.16 Consider the following C function.

```
int fun (int n)
{
    int x=1, k;
    if (n==1) return x;
    for (k=1; k<n; ++k)
    x = x + fun(k) * fun(n – k);
    return x;
}
```

The return value of fun(5) is _____.

(A) 0 (B) 26 (C) 51 (D) 71

Q.17 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

Q.18 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 _____ .

Q.19 Consider the C functions foo and bar given below :

```
int foo (int val)
{
    int x = 0;
    while (val > 0)
```

```
        {
         x = x + foo (val – –);
        }
    return val;
}
int bar (int val)
{
    int x =0;
    while (val > 0)
    {
        x = x + bar (val –1);
    }
    return val;
}
```

Invocations of foo (3) and bar (3) will result in :

(A) Return of 6 and 6 respectively.

(B) Infinite loop and abnormal termination respectively.

(C) Abnormal termination and infinite loop respectively.

(D) Both terminating abnormally.

Q.20 What is the output of the following C code? Assume that the address of X is 2000 (in decimal) and an integer requires four bytes of memory.

```
int main ()  {
unsiged int x[4] [3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12}};
printf("%u, %u,%u", x + 3, *(x +3),  *(x +2) +3);   }
```

(A) 2036, 2036, 2036

(B) 2012, 4, 2204

(C) 2036, 10, 10

(D) 2012, 4, 6

11 →

```
int unknown (int n)
{
    int i, j, k = 0;
    for (i = n/2 ; i <= n ; i++)
        for (j = 2; j <= n ; j = j * 2)
            k = k + n/2 ;
    return (k);
}
```

$(\frac{n}{2}+1)$

$\log n$

$n^2 \log n$

n $n \log n$ for inner loop

$n^2 \log n$

Value of f^n = $n^2 \log n$

12 →

```
# include <stdio.h>
void main()
{
    int a = 3;
    if (a--, a--, a--, a)
        printf ("Pankaj");
    else
        printf ("Sharma");
}
```

a = 2
a = 1
a = 0

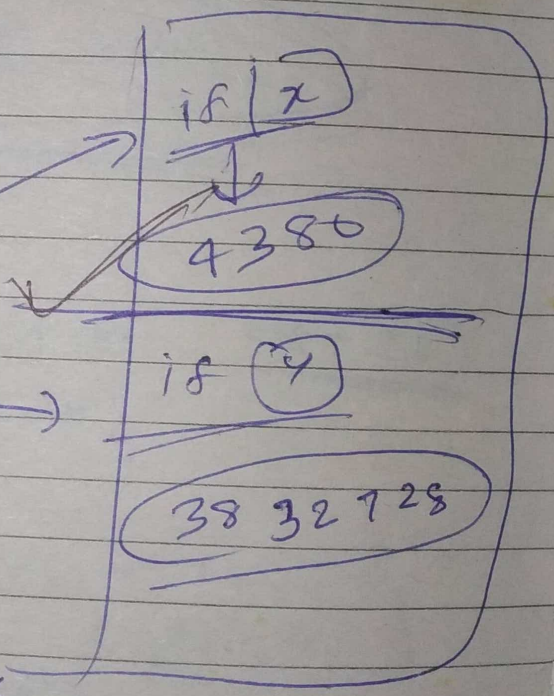
Output - Sharma

13⇒

```
#include <stdio.h>
int funcf(int x);
int funcg(int y);
main()
{
  int x=5, y=10, count;
  for (count=1; count<=2; ++count)
  {
    y += funcf(x) + funcg(x);
    printf("%d", y);
  }
}

funcf(int x)
{
  int y;
  y = funcg(x);
  return(y);
}

funcg(int x)
{
  static int y=10;
  y++; y+=1;
  return(y+x);
}
```



Output - ~~38 32 728~~ 43 80

(14)

```
int a, b, c = 0;
void ptrFun(void);
int main()
{
    static int a = 1;
    ptrFun();
    a += 1;
    ptrFun();
    printf("%d %d", a, b);
}

void ptrFun(void)
{
    static int a = 2;
    int b = 1;
    a += ++b;
    printf("\n %d %d", a, b);
}
```

Output :-

4 2
6 2
2 0

15)

```
#include <stdio.h>
int r()
{
    static int num = 7;
    return num--;
}

int main()
{
    for ( r(); r(); r())
        printf("%d", r());
    return 0;
}
```

output → 52

16)

```
int fun (int n)
{ int x=1, k;
    if (n==1) return n;
    for (k=1; k<n; ++k)
        x = x + fun(k) * fun(n-k);
    return x;
}
```

fun(5) = 51

(17)

```
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);
```

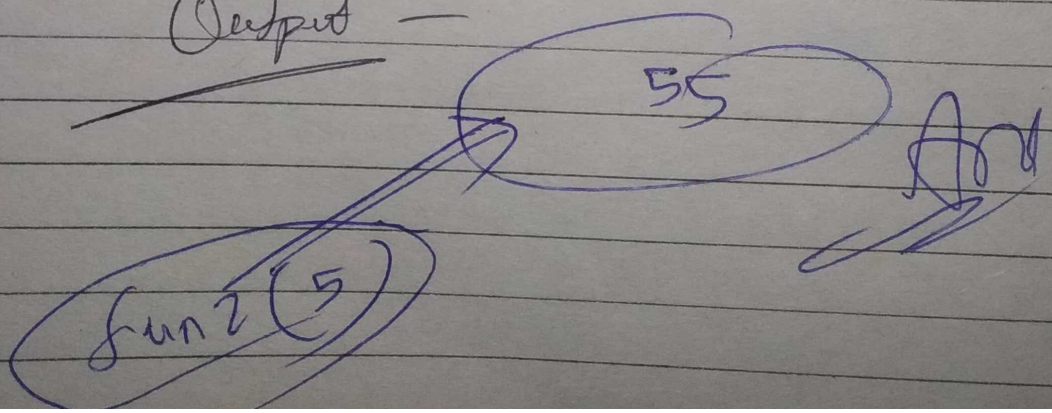
fun1(5) = 5342312223345

(18)

```
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);
}
```

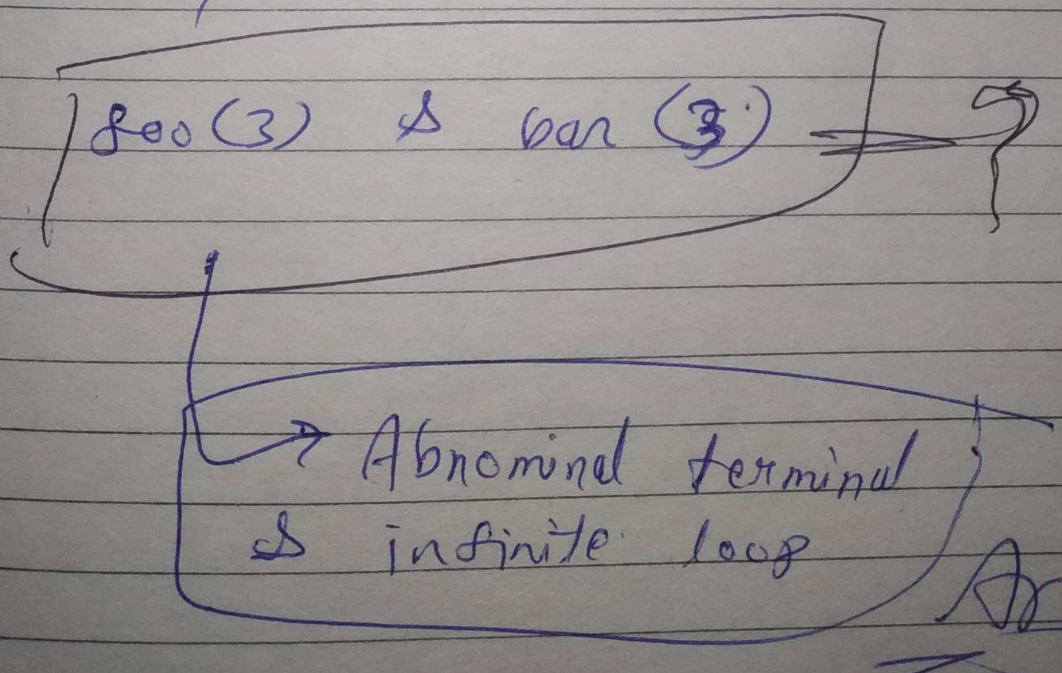
Output —



(19)

```
int foo(int val)
{
    int x = 0;
    while (val > 0)
    {
        x = x + foo(val--);
    }
    return val;
}

int bar(int val)
{
    int x = 0;
    while (val > 0)
    {
        x = x + bar(val-1);
    }
    return val;
}
```



Ans

(2035)

```
int main()
{
    unsigned int x[4][3] = {{1,2,3},
                            {4,5,6},
                            {7,8,9},
                            {10,11,12}};

    printf("%u %u %u %u", x+3, *(x+3), *(x+2)+3);
}
```

→ output

→ 2036, 2036, 2036