CS & IT ENGINEERING

Programming in C

Functions and Storage Classes

Lec-07

By- Pankaj Sharma sir

Recursion - 3

Q.1

```
void f(int n)
   {
       if(n<=0)
       return;
       printf("%d",n);
       f(n-1);
   }
```

What is the output of f(5)

Q.2

```
void f(int n)
   {
       if(n<=0)
       return;
       f(n-1);
       printf("%d",n);
   }
```

What is the output of f(5)

Q.3

```
void f(int n)
   {
       if(n<=0)
       return;
       f(n-1);
       printf("%d",n);
       f(n-1);
   }
```

What is the output of f(4)

Q.4

```
int f(int n)
    {
        if(n<=1)
        return n;
        return f(n/2) + f(n/2) + 1;
    }
```

What is the output of f(5)

Q.5

```
int f(int n)
    {
        if(n<=1)
        return n;
        return f(n/2) + n/2 ;
    }
```

What is the output of f(12)

Q.6

```
int f(int n)
    {
        if(n<=1)
        return n;
        if(n%2)
        return f(n/2) + n;
        return f(n/3) + n;
    }
```

output of f(22) ?

Q.7

Consider the code :

```
/* Assume that n>=0 */
void fun(int n)
   {
      if(n==0)
      return 0;
      fun(n/2);
      printf("%d",n%2);
   }
```

output of f(11)?

Q.8

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);
}
void main(){
int a=2018,sum=0;
foo(a,sum);
printf("%d",sum);
}
```

Output?

A. 8, 4, 0, 2, 14

B. 8,4,0,2,0

C. 2,0,4,8,14

D. 2,0,4,8,0

Q.9

```
void main()
{
    static int var=5;
    printf("%d",var--);
    if(var)
    main();
}
```

Q.10

```
void main()
 {
    static int i =5;
     if(--i)
        {
          main();
          printf("%d",i);
        }
 }
```

Q.11

predict the output

```
int fun(int x)
  {
     if(x%2==0)
     return fun(fun(x-1));
  else
     return(x++);
  }
int main()
  {
     printf("%d",f(12));
     getchar();
     return 0;
  }
```

A. 10

B. 11

C. 12

D. None of these

Q.12

```
int fun(int a,int b)
   {
      if(b==0)
      return 0;
      if(b%2==0)
      return fun(a+a,b/2);
      return fun(a+a,b/2) + a;
   }

int main()
   {
      printf("%d",fun(4,3));
      getchar();
      return 0;
   }
```

A. 12

B. 81

C. 64

D. 8

Q.13

Consider the following C function :

```
int f(int n)
 {
   static int r=0;
   if(n<=0)
   return 1;
   if(n<3)
   {
      r=n;
      return f(n-2) + 2;
   }
   return f(n-1) + r;
 }
```

what is the value of f(5)

A. 5

B. 7

C. 9

D. 18

Q.14

Consider the following recursive C function

```
unsigned int foo(unsigned int n, unsigned int r)
 {
    if(n>0)
    return (n%r) + foo(n/r , r);
else
    return 0;
 }
```

output of foo(513,2)

A. 9

B. 8

C. 5

D. 2

Q.15 Which of the following statements is/are valid?

A. return a+b;

B. return a,b,c;

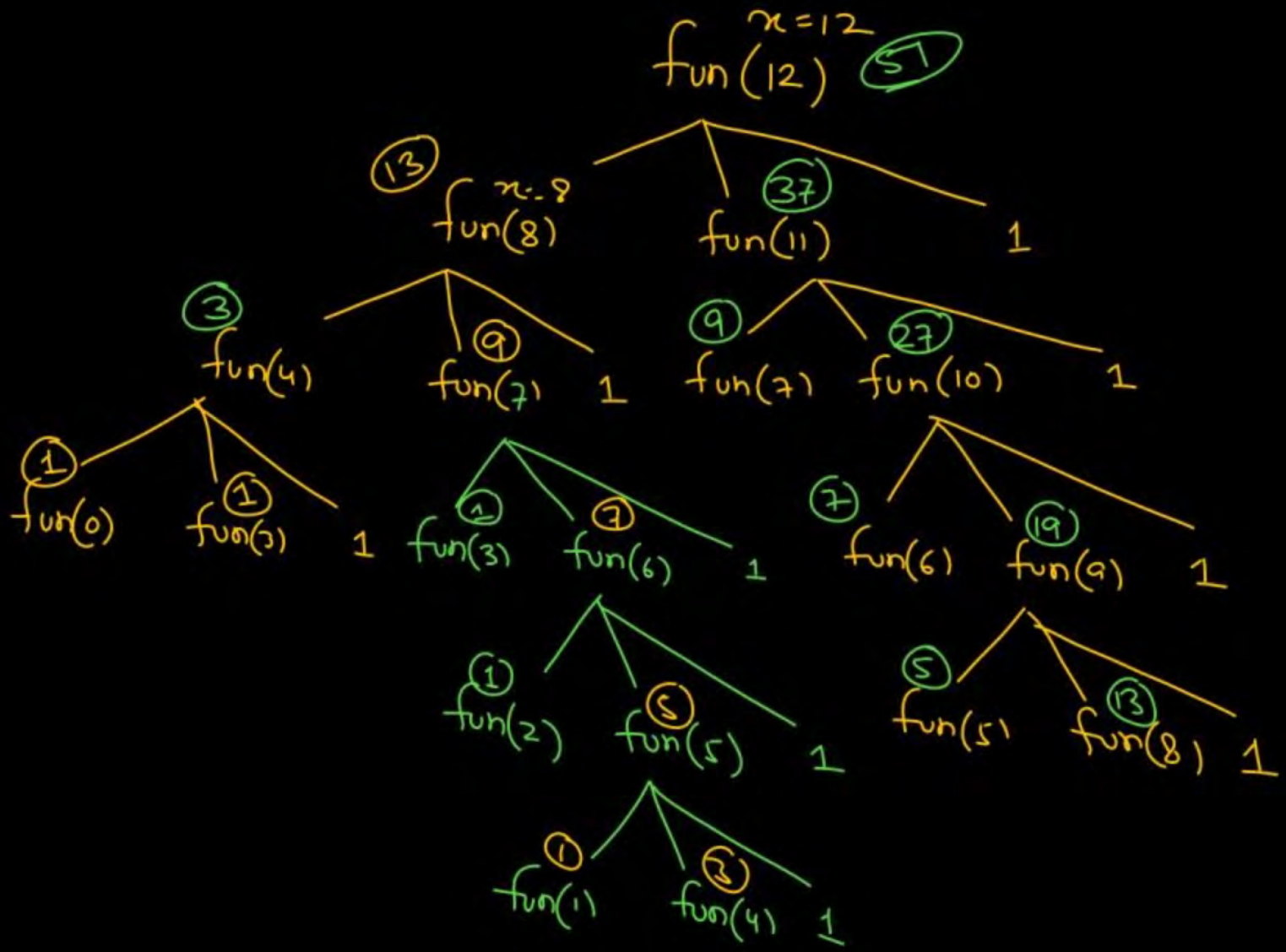
C. return (a,b,c);

D. All of them

Q.16

```
int fun(int x)
 {
   if(x>3)
   return fun(x-4) + fun(x-1) + 1;
   return 1;
 }
```

Find the value returned by fun(12)


x=12
fun(12)
57
13
x=8
fun(8)
37
fun(11)
1
3
fun(4)
9
fun(7)
1
9
fun(7)
27
fun(10)
1
1
fun(0)
1
fun(1)
1
1
fun(3)
7
fun(6)
1
7
fun(6)
19
fun(9)
1
1
fun(2)
5
fun(5)
1
5
fun(5)
13
fun(8)
1
1
fun(1)
3
fun(4)
1

$fun(0) = 1$

$fun(1) = 1$

$fun(2) = 1$

$fun(3) = 1$

$fun(4) = fun(0) + fun(3) + 1 = 3$

$fun(5) = fun(1) + fun(4) + 1 = 5$

$fun(6) = fun(2) + fun(5) + 1 = 7$

$fun(7) = fun(3) + fun(6) + 1 = 9$

$fun(8) = fun(4) + fun(7) + 1 = 13$

$fun(9) = fun(5) + fun(8) + 1 = 19$

$fun(10) = fun(6) + fun(9) + 1$

$= 7 + 19 + 1 = 27$

Q.17

Predict output of following program

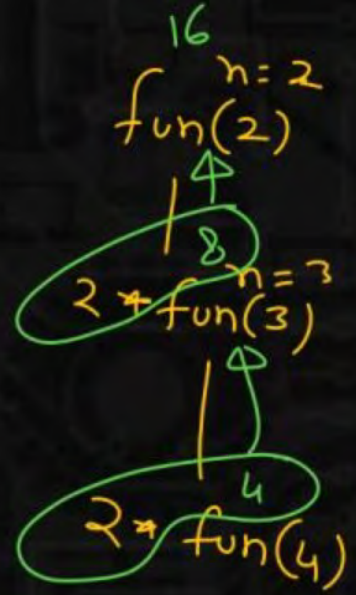
```
#include <stdio.h>
int fun(int n)
{
   if (n == 4)
   return n;
   else return 2*fun(n+1);
 }
int main()
 {
   printf("%d ", fun(2));
   return 0;
 }
```

A. 4

B. 8

C. 16

D. error



Q.18

Consider the following recursive function fun(x, y). What is the value of fun(4, 3)

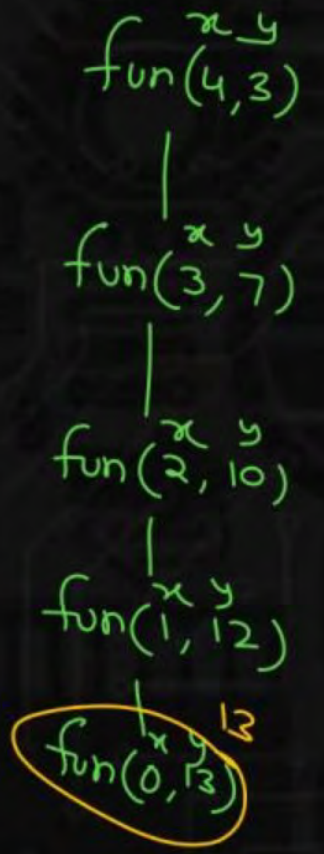
```
int fun(int x, int y)
{
   if (x == 0)
   return y;
   return fun(x - 1,  x + y);
}
```

A. 13

B. 12

C. 9

D. 10



Q.19

What does the following function do?

```
int fun(int x, int y)
{
  if (y == 0)   return 0;
  return (x + fun(x, y-1));
}
```

A. x + y

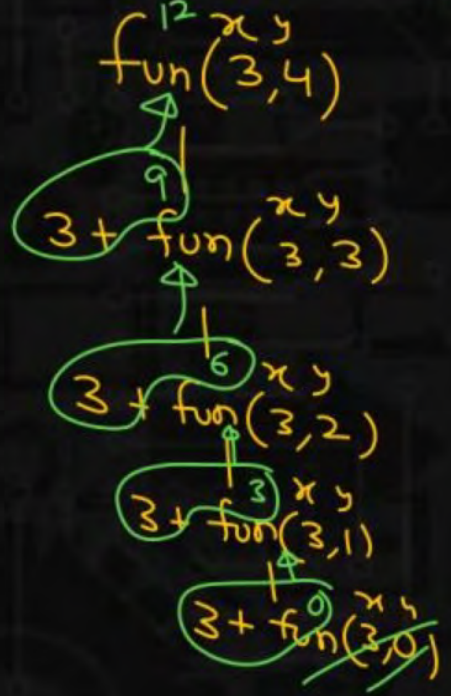
B. x + x * y

C. x * y

D. pow(x, y)

fun(x,y) =

fun(3,4) = 12



Q.20

What does fun2() do in general?

3-4 min

```
int fun(int x, int y)
{
    if (y == 0)   return 0;
    return (x + fun(x, y-1));
}
```

A. x * y

B. x + x * y

C. pow(x, y)

D. pow(y, x)

fun(x, y) = x * y

```
int fun2(int a, int b)
{
    if (b == 0) return 1;
    return fun(a, fun2(a, b-1));
}
```

Q.20

What does fun2() do in general?

3-4 min

```
int fun(int x, int y)
{
   if (y == 0)   return 0;
   return (x + fun(x, y-1));
}
```

A. x * y

B. x + x * y

C. pow(x, y)

D. pow(y, x)

fun(x, y) = x * y

```
int fun2(int a, int b)
{
   if (b == 0) return 1;
   return fun(a, fun2(a, b-1));
}
```

fun2(a, b)

a * fun2(a, b-1)

a * fun2(a, b-2)

a * fun2(a, b-3)

Q.21

Output of following program?

3 min

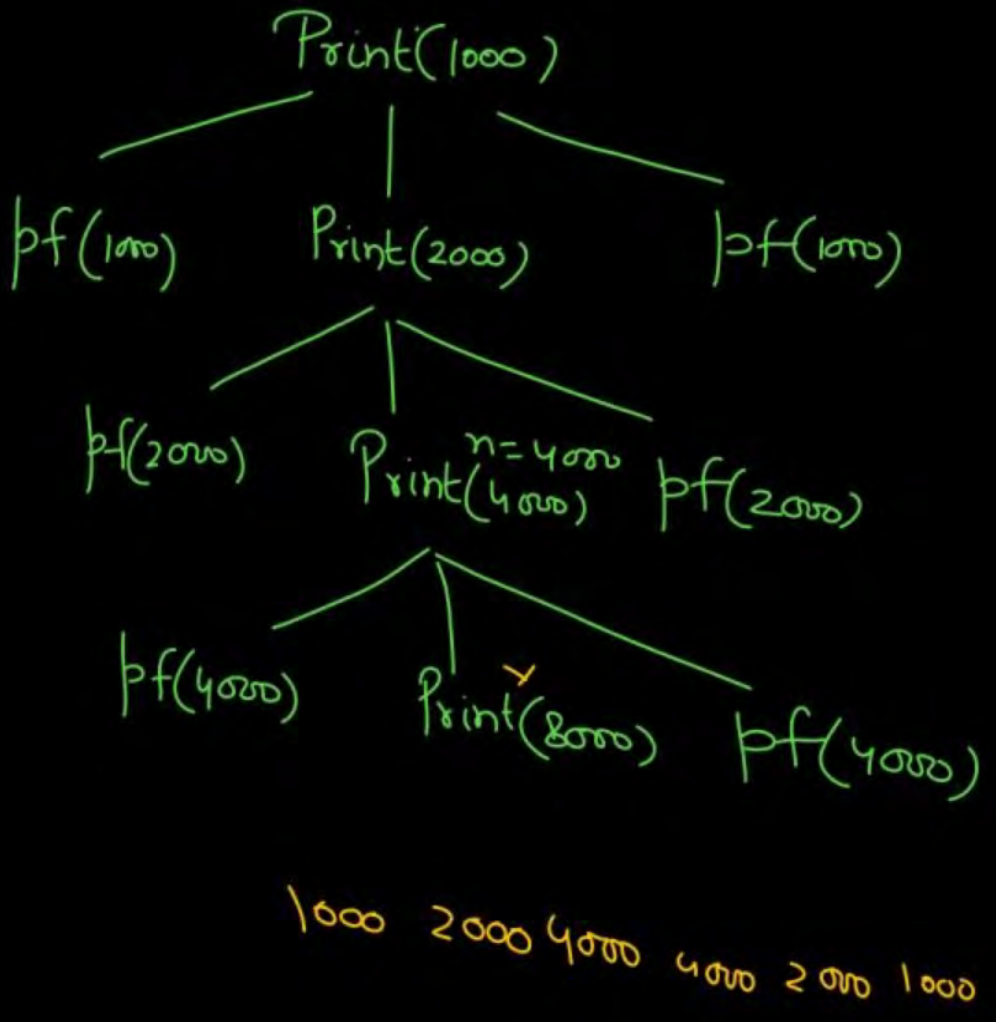
```
#include<stdio.h>
void print(int n){
   if (n > 4000)
   return;
   printf("%d ", n);
   print(2*n);
   printf("%d ", n);
}
int main()
{
   print(1000);
   getchar();
   return 0;
}
```

A. 1000 2000 4000

B. 1000 2000 4000 4000 2000 1000

C. 1000 2000 4000 2000 1000

D. 1000 2000 2000 1000


Print(1000)
pf(1000)
Print(2000)
pf(1000)
pf(2000)
Print(4000)
n=4000
pf(2000)
pf(4000)
Print(8000)
pf(4000)
1000 2000 4000 4000 2000 1000

Q.22

What does the following function do?

```
int fun(unsigned int n)
{
   if (n == 0 || n == 1)
      return n;

   if (n%3 != 0)
      return 0;
   return fun(n/3);
}
```

fun(1) = 1

b<3 fun(1)=1

A. It returns 1 when n is a multiple of 3, otherwise returns 0

B. It returns 1 when n is a power of 3, otherwise returns 0

C. It returns 0 when n is a multiple of 3, otherwise returns 1

D. It returns 0 when n is a power of 3, otherwise returns 1

Q.22

What does the following function do?

```
int fun(unsigned int n)
{
  if (n == 0 || n == 1)
    return n;

  if (n%3 != 0)
    return 0;
  return fun(n/3);
}
```

fun(1) = 1

fun(6) = fun(2) = 0

not div by 3

A. It returns 1 when n is a multiple of 3, otherwise returns 0

B. It returns 1 when n is a power of 3, otherwise returns 0

C. It returns 0 when n is a multiple of 3, otherwise returns 1

D. It returns 0 when n is a power of 3, otherwise returns 1

Q.22

What does the following function do?

```
int fun(unsigned int n)
{
    if (n == 0 || n == 1)
        return n;

    if (n%3 != 0)
        return 0;
    return fun(n/3);
}
```

not div by 3

fun(1) = 1

fun(6) = fun(2) = 0

fun(5) = 0

fun(4) = 0

fun(9) = fun(3) = fun(1) = 1

$fun(3^k) = fun(3^{k-1}) = fun(3^{k-2})$

$= \cdots fun(3^1)$

$= fun(1)$

$= 1$

A. It returns 1 when n is a multiple of 3, otherwise returns 0

B. It returns 1 when n is a power of 3, otherwise returns 0

C. It returns 0 when n is a multiple of 3, otherwise returns 1

D. It returns 0 when n is a power of 3, otherwise returns 1

Q.22

What does the following function do?

```
int fun(unsigned int n)
{
    if (n == 0 || n == 1)
        return n;

    if (n%3 != 0)
        return 0;
    return fun(n/3);
}
```

fun(1) = 1

fun(2) = 0
fun(3) = fun(1) = 1
fun(4) =

$fun(3^k) = fun(3^{k-1}) = fun(3^{k-2})$
$= \cdots fun(3^1)$
$= fun(1)$
$= 1$

A. It returns 1 when n is a multiple of 3, otherwise returns 0

B. It returns 1 when n is a power of 3, otherwise returns 0

C. It returns 0 when n is a multiple of 3, otherwise returns 1

D. It returns 0 when n is a power of 3, otherwise returns 1

Q.23

Predict the output of following program

```
#include <stdio.h>
int f(int n)
{
   if(n <= 1)
      return 1;
   if(n%2 == 0)
      return f(n/2);
   return f(n/2) + f(n/2+1);
}
 int main()
{
   printf("%d", f(11));
   return 0;
}
```

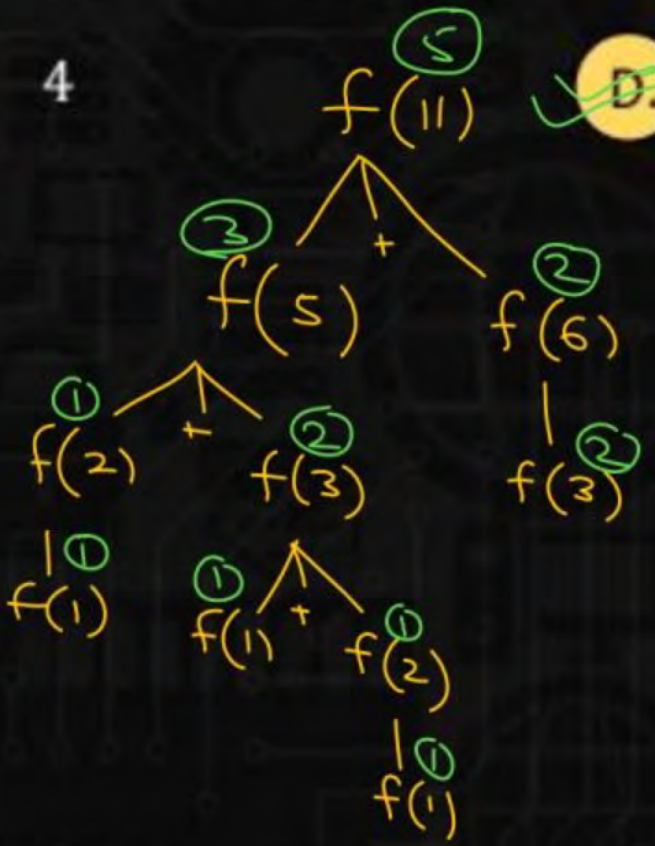
n is even

A. Stack Overflow

B. 3

C. 4

D. 5



Q.24 Consider the following C function:

```
int f(int n)
{
  static int i = 1;
  if (n >= 5)
    return n;
  n = n+i;
  i++;
  return f(n);
}
```

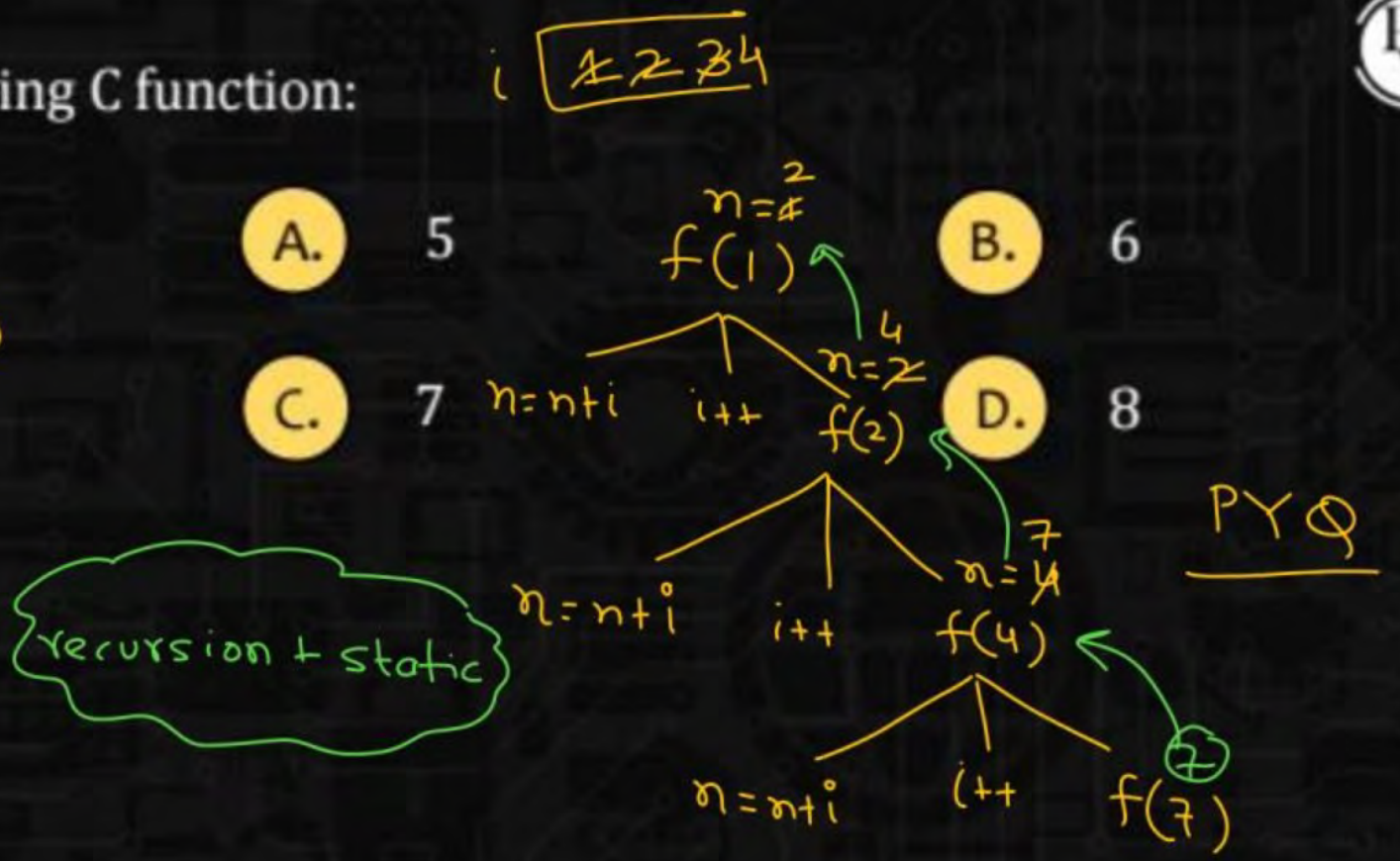
The value returned by f(1) is

A. 5

B. 6

C. 7

D. 8



Q.25 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 _____.

H.W

Pankaj sir

A. 0

B. 26

C. 51

D. 71

PYQ

Q.26

Consider the following recursive C function. If get(6) function is being called in main() then how many times will the get() function be invoked before returning to the main()?

```
void get (int n)
{
  if (n < 1) return;
1. get(n-1);
2. get(n-3);
  printf("%d", n);
}
```

get(0), get(-1), get(-2) ... ⟹ No further call

PYQ

Every rec. call

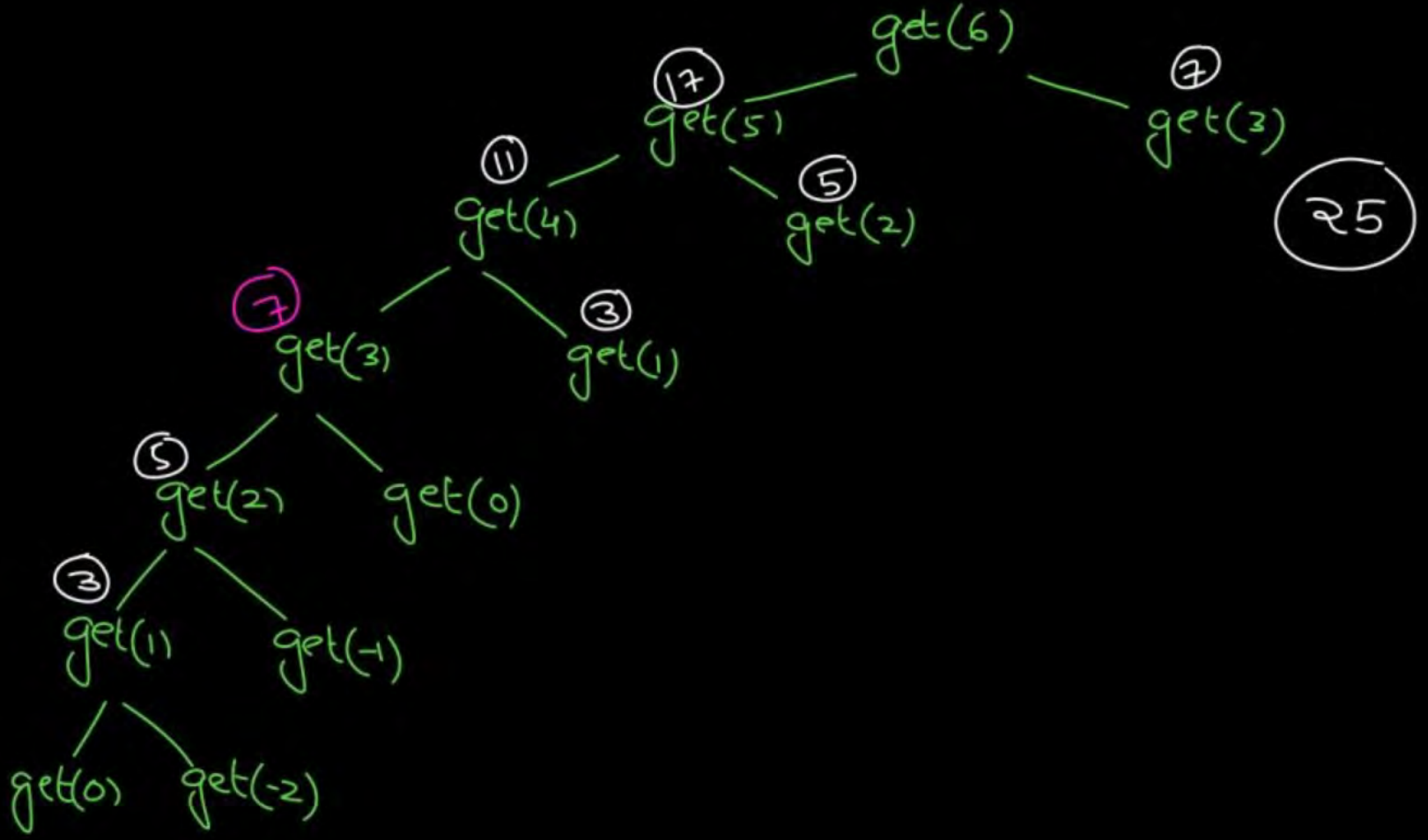
Easy ⟹

A. 15

B. 25

C. 35

D. 45


get(6)
17
get(5)
7
get(3)
11
get(4)
5
get(2)
25
7
get(3)
3
get(1)
5
get(2)
get(0)
3
get(1)
get(-1)
get(0)
get(-2)

Q.27 What will be the output of the following C program?

PYQ

```
void count(int n)
{
    static int d = 1;
    printf("%d ", n);
    printf("%d ", d);
    d++;
    if(n > 1) count(n-1);
    printf("%d ", d);
}
int main()
{
    count(3);
}
```

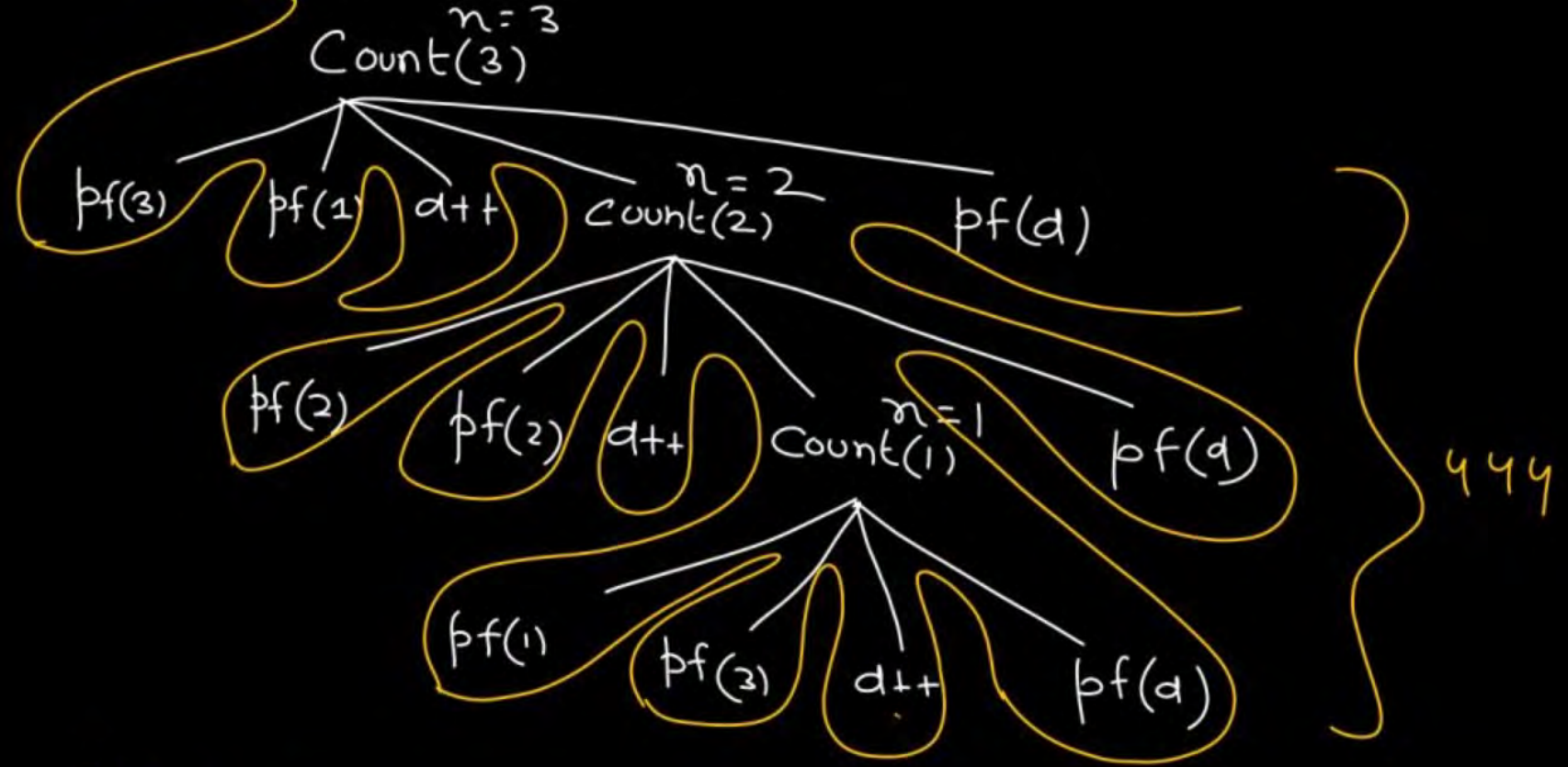
A. 3 1 2 2 1 3 4 4 4

B. 3 1 2 1 1 1 2 2 2

C. 3 1 2 2 1 3 4

D. 3 1 2 1 1 1 2

d 1234



312 213 444

Q.28

What will be the output of the C program?

```
#include<stdio.h>
int main()
{
  function();
  return  0;
}
void function()
{
  printf("Function in C is awesome");
}
```

A. Function in C is awesome

B. no output

C. Runtime error

D. Compilation error

Q.29

What will be the output of the C program?

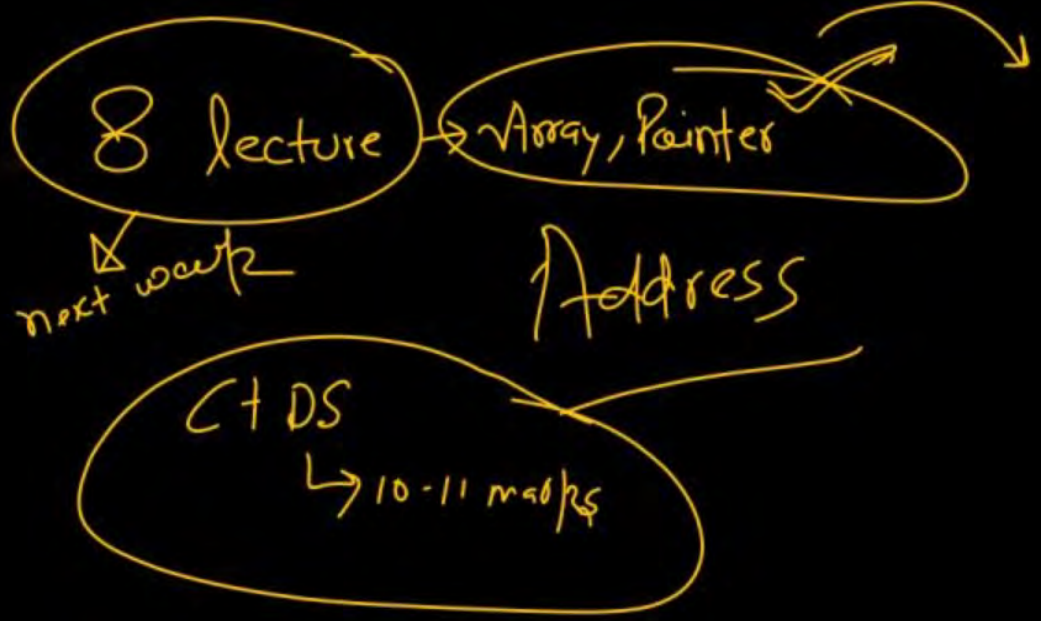
```
#include<stdio.h>
int main()
{
   main();
   return 0;
}
```

A. Runtime error

B. Compilation error

C. 0

D. None of these


8 lecture
Array, Pointer
next week
Address
C+DS
10-11 marks


THANK YOU GW
SOLDIERS !