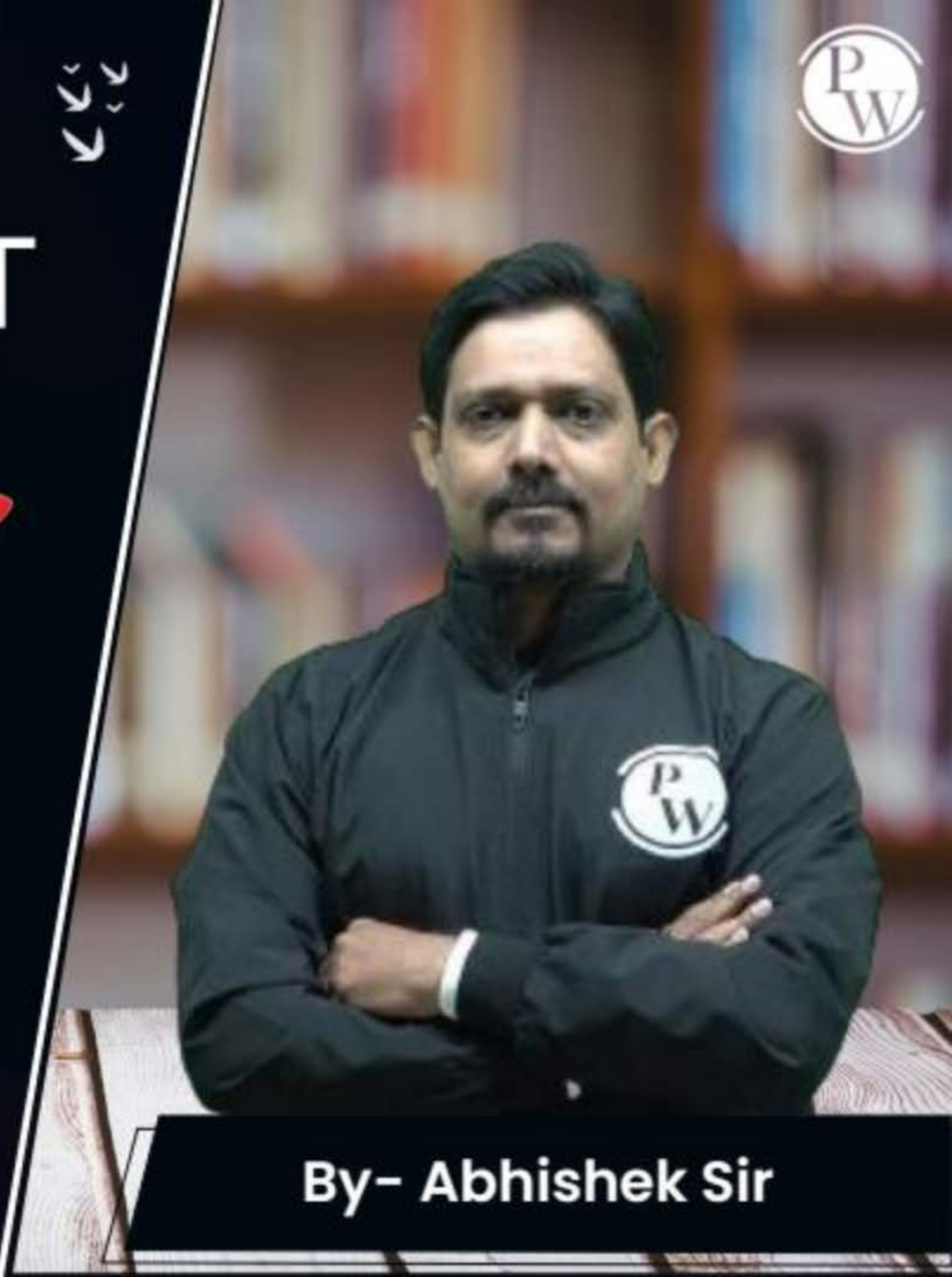
Computer Science & IT

C programming

Functon & Storage Class

Lecture No. 01



By- Abhishek Sir

Recap of Previous Lecture

- Topic — do while
- Topic — Break
- Topic — continue
- Topic
- Topic

Slide

Topics to be Covered

Topic

Topic

Topic

Topic

Topic

function

Activation Record

Activation Tree

1. function
   Activation Record/Tree
2. Storage class
3. Recursion
4. Recursion

Slide

Function

- What is a function
- Why function
- Function execution
- Function prototype
- Types of function
- Declaration & definition
- Function call
- Parameter (formal parameter), argument (Actual parameter)
- Return value

Group of Instruction from a logical unit

Modularity / Reusability

Control transfer

return type Name of function(parameter)

Function

```
#include <stdio.h>
int fun(int, int);
int main(){
 int a = 10, b = 20;
 printf("%d", fun(a,b));
}
int fun(int x,int y){
   int z;
   z = x++* ++y;
   return z;
}
```

function declaration / return type. Name (parameter)

function definition

```
int main () {
}
```

function definition

Slide

Function

```
#include <stdio.h>
int fun(int, int);
int main(){
 int a = 10, b = 20;
 printf("%d", fun(a,b));
}
int fun(int x,int y){
   int z;
   z = x++* ++y;
   return z;
}
```

function call

1 control transfer

2 a, b argument
actual parameter

x, y parameter
formal parameter

3 formal parameter copied with actual parameter
x:a, y:b
Call by value

4 function executes, value returned and control transfer back to main function
Callee : Main
Called fun

Slide

Question

#Q

```
double foo (double);              /* Line 1 */
int main() {
    double da, db;
    //input da
    db = foo(da);
}
double foo (double a) {
    return a;
}
```

Assume that function will return integer

Best Answer [D]

The above code compiled without any error or warning. If Line 1 is deleted, the above code will show:

A. no compile warning or error

B. some compiler-warnings not leading to unintended results

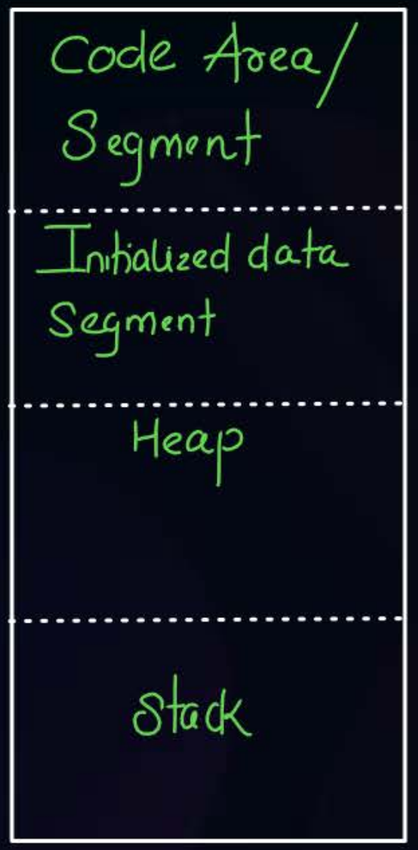
C. some compiler-warnings due to type-mismatch eventually leading to unintended results

D. compiler errors

Slide

Function

How function call implemented ??



Memory Layout of program

using Stack function calls are implemeted

Slide

Activation Tree and Records

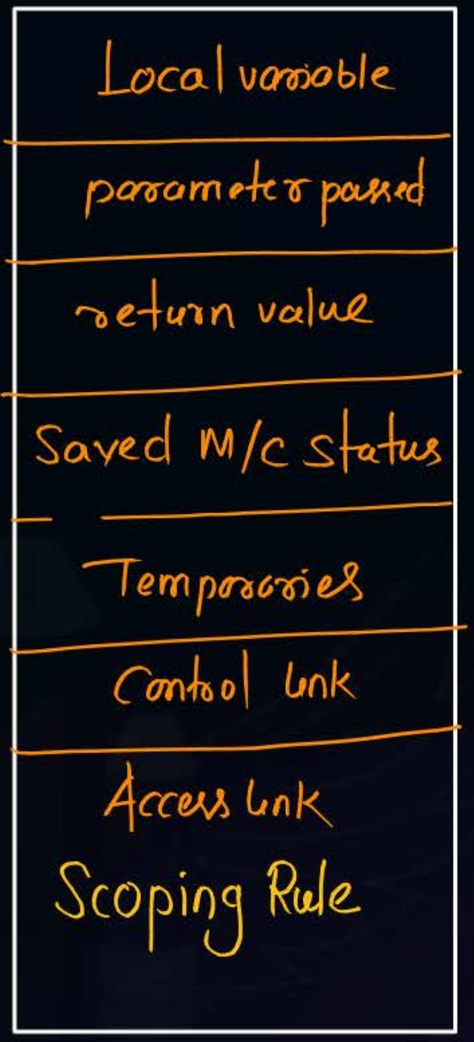
$a = 5 \times 10 / 3$

Activation

function is in execution

Activation Record :- When function is in execution then all information about function kept as Record structure called Activation Record.

As function start executing then Activation Record is created and pushed in Run time stack and upon termination popped from Run time stack



Activation Record

Slide

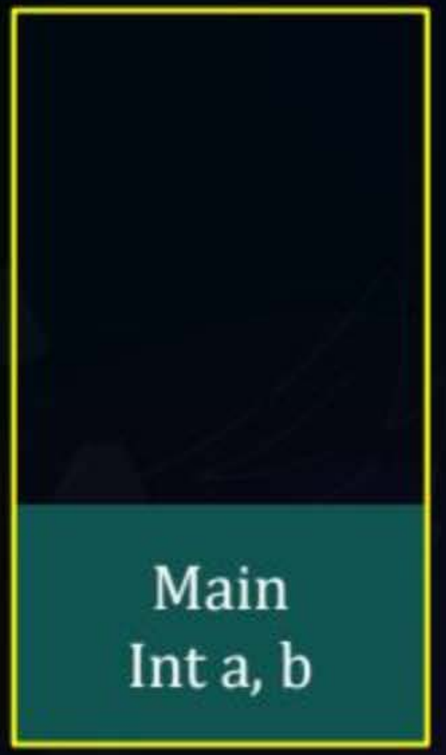
Activation Tree and Records

Example: Consider the following program

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```



Slide

Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack

Slide

Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack

F2() Int c , d
F1() Int b,c
Main Int a, b

Slide

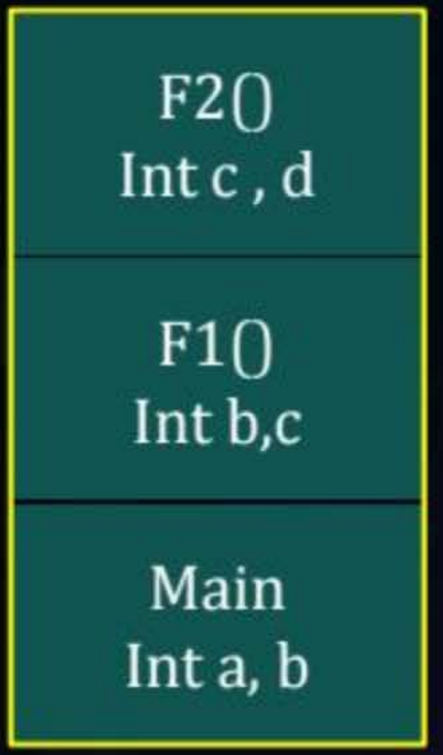
Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack



Slide

Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack

Slide

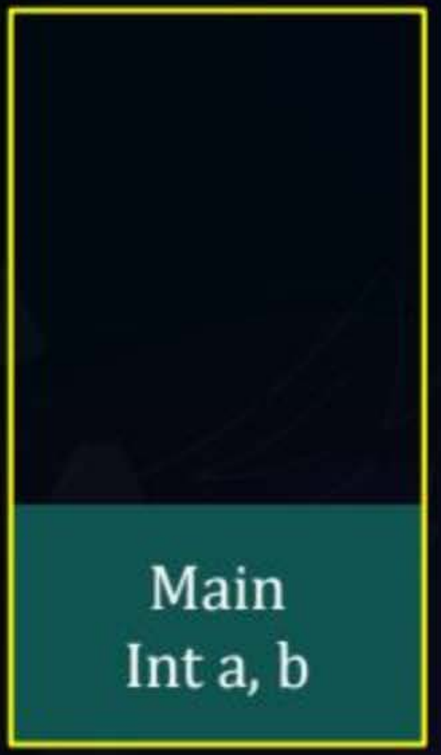
Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack



Slide

Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack

Slide

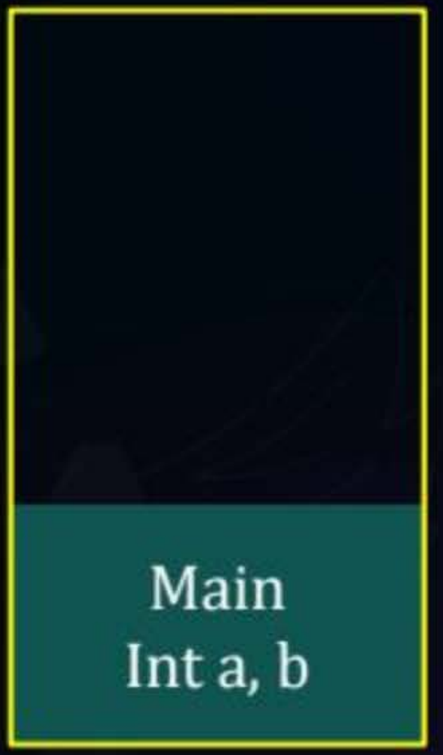
Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack



Slide

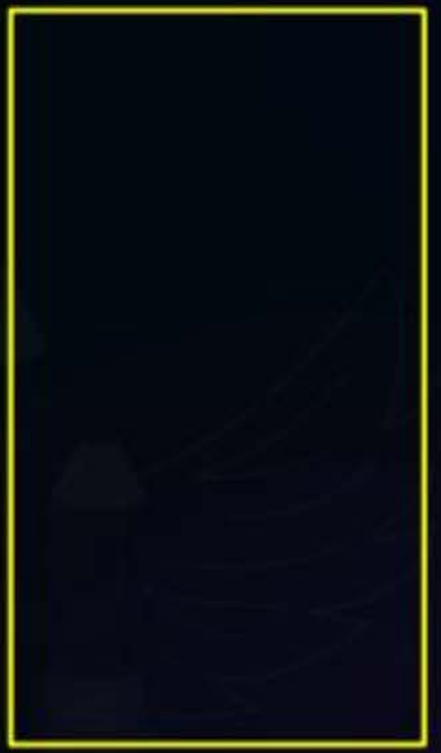
Activation Tree and Records

```
int main(){
    int a,b;
    .
    .
    .
    f1();
    f2();
}
```

```
void f1(){
    int b,c;
    .
    .
    .
    f2();
}
```

```
Void f2(){
    int c,d;
    .
    .
    .
}
```

Stack



Slide

Question

X : 22.5

X 22

default - 22 times

#Q

No. of times '*' will be printed by the following C code is _____

```
#include<stdio.h>
void foo(int x){
        for(int i =1; i<=x; i++){

        switch(x){
                case 1: printf("*");i+=2;
                case 2: printf("*");
                case 3: printf("*");
                default: printf("*");
        }
        }
}
int main()
{
        foo(22.5);
}
```

Number of star printed is ?

(A) 21

(B) 22

(C) 23

(D) 24

Slide

Question

#Q

No. of times '*' will be printed by the following C code is _____

```
#include<stdio.h>
void foo(int x){
        for(int i =1; i<=x; i++){

        switch(x){
                case 1: printf("*");i+=2;
                case 2: printf("*");
                case 3: printf("*");
                default: printf("*");
        }
        }
}
int main()
{
        foo(22.5);
}
```

Number of star printed is ?

(A) 21

(B) 22

(C) 23

(D) 24

X: 22.5

X 22

i = 1 - 4 i = 3

i++ i = 4 default

i = 22.

22 - 4 + 1 = 19

19 + 4 = 23

Slide

Question

#Q The number of character printed by the code _____

```
#include<stdio.h>
void a();
void b();
void c();
int main(){
      a();
      b();
      return 0;
}
void a(){ printf("a"); b();}
void b(){ printf("a"); c();}
void c(){ printf("a");}
```

(A) 2

(B) 3

(C) 4

(D) 5

Slide

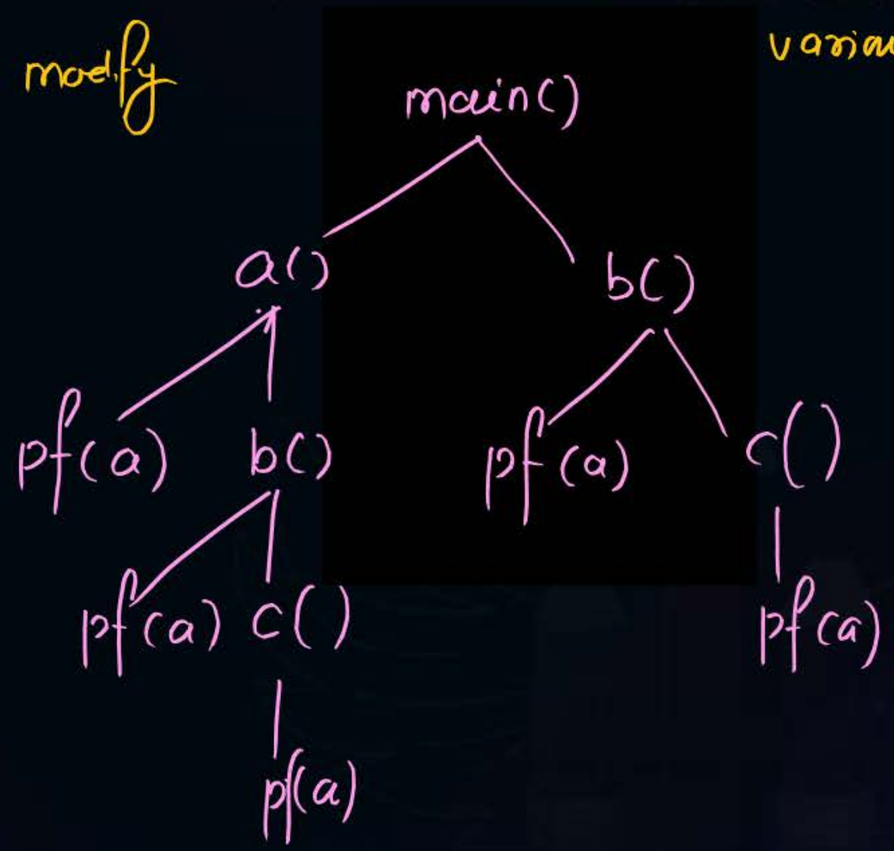
Question

#Q The number of character printed by the code

```
#include<stdio.h>
void a();
void b();
void c();
int main(){
      a();
      b();
      return 0;
}
void a(){ printf("a"); b();}
void b(){ printf("a"); c();}
void c(){ printf("a");}
```

Activation Tree - modify

Activation Tree : control transfer between various functions



Slide

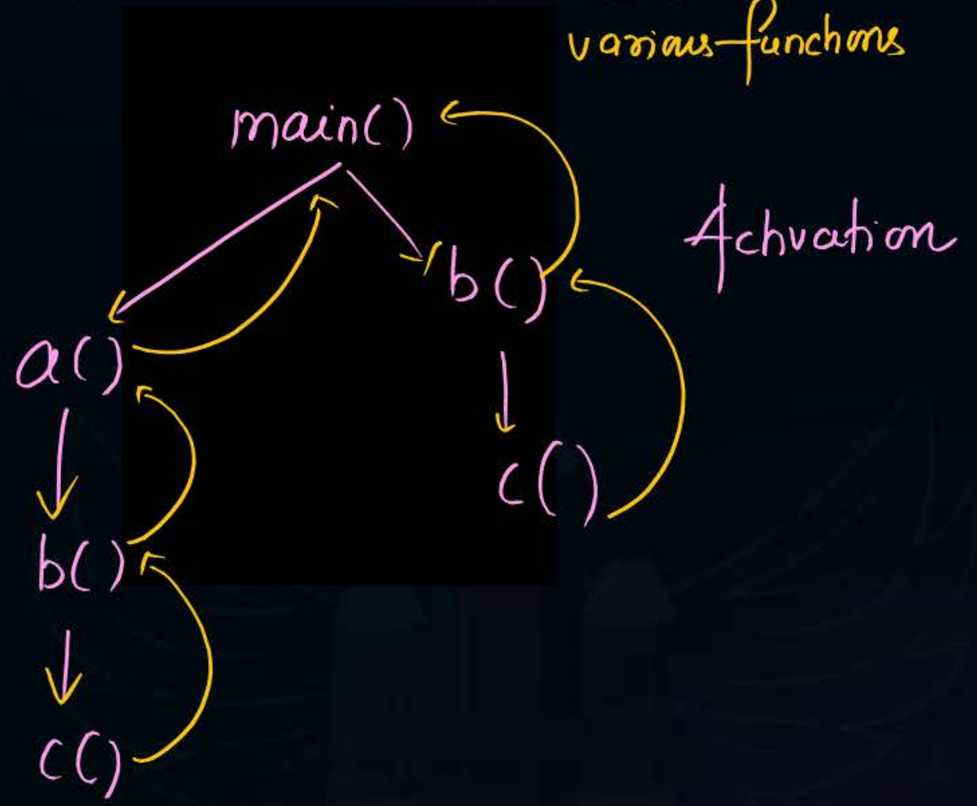
Question

#Q The number of character printed by the code

```
#include<stdio.h>
void a();
void b();
void c();
int main(){
      a();
      b();
      return 0;
}
void a(){ printf("a"); b();}
void b(){ printf("a"); c();}
void c(){ printf("a");}
```

Achvahon Tree - modify

Achvahon Tree : control transfer between various functions



Slide

Question

#Q Consider the following program

```
#include <stdio.h>
void a(){printf("1");}
void b(){a();printf("2");}
void c(){a();b();printf("3");}
void d(){a();b();c();printf("4");}

int main(){
   d();
}
```

What is the output of the following program?

(A) 11211234

(B) 11211243

(C) 11212134

(D) 11211324

[A]

Slide

Question

output

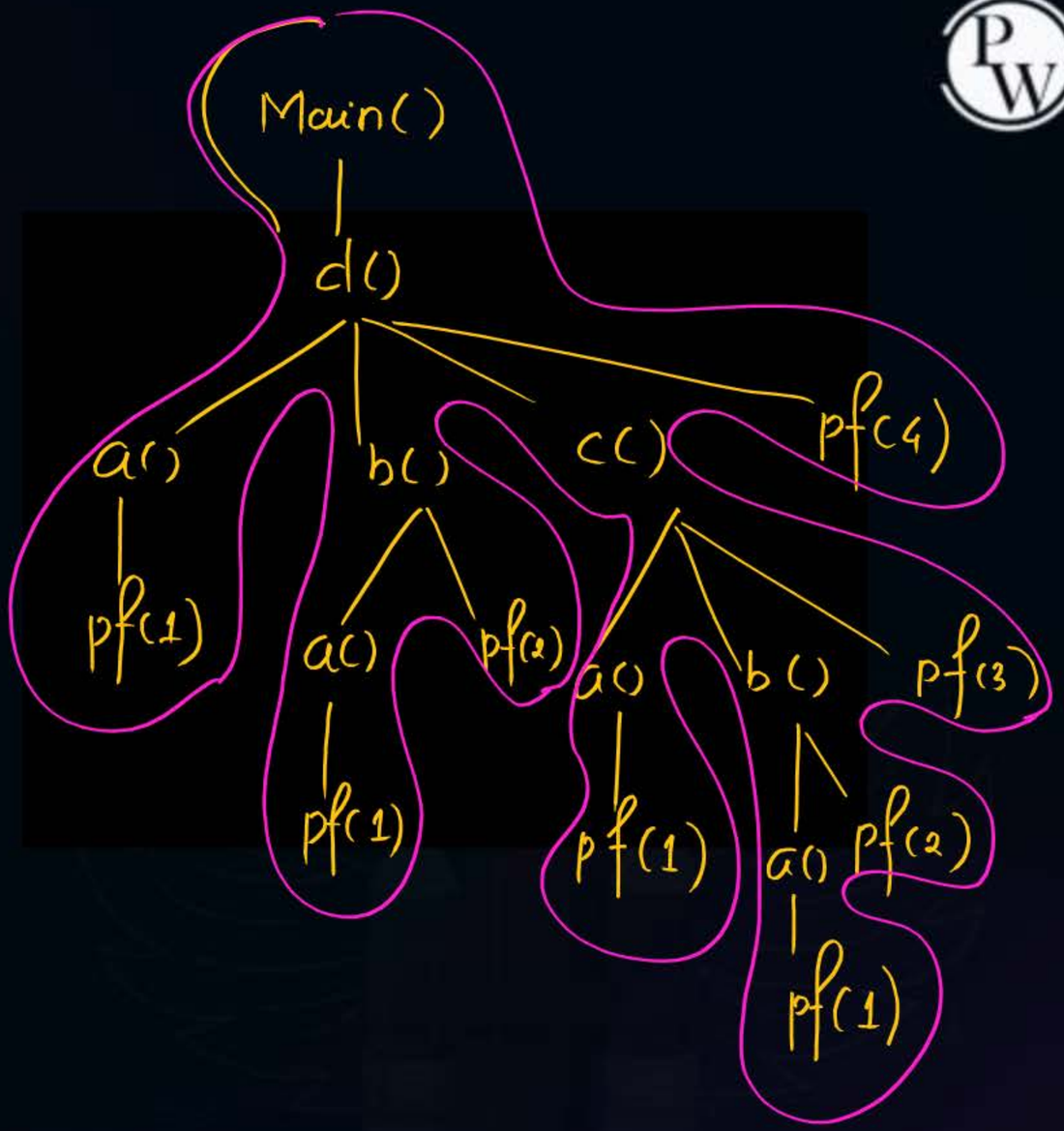
tracing

#Q Consider the following program

```
#include <stdio.h>
void a(){printf("1");}
void b(){a();printf("2");}
void c(){a();b();printf("3");}
void d(){a();b();c();printf("4");}

int main(){
   d();
}
```

1 12 1123 4



Slide

Question

```
#include<stdio.h>
int foo(int M){
            int i =0;
            while(i*i < M)
                  i++;
      return i;
      }

int main(){
    printf("%d", foo(10)+foo(30)+foo(60));
}
```

The value printed by the program is _____

foo(10) $3^2 < 10$, true - 4 $4^2 < 10$

foo(30) $5^2 < 30$, - 6 $6^2 < 30$

foo(60) $7^2 < 60$ - 8 $8^2 < 60$

4 6 8

Slide

Question

```
#include<stdio.h>
int bar(int x, int n){
        int value =1;
        if(n>0){
            if(n%2==1)
                value = value*x;
            value = value*bar(x*x, n / 2);
        }
        return value;
    }
int main(){
    printf("%d", bar(3,5)+bar(2,5)+bar(6,3));
    }
```

243
32
216
491

bar(3,5) × n

value=3

value: 3 * bar(9,2)

81

81

value: 1 * bar(81,1)

81*1=81

value: 81

value: 81*

bar(81*81, 0)

1

3^5 = 243 32 216

The output of the program is 491

Slide

Question

#Q

Consider the following C program:

```
#include<stdio.h>
int f1(void);
int f2(void);
int f3(void);
int f4(int, int, int, int);
int x=10;
int main(){
    int x=1;
    x+= f4(f1(), f2() , f3() , f2());
    printf("%d", x);
    return 0;
}
```

The output of the program is _____.

```
int f1() { int x = 25; x++; return x;}
int f2() {int x = 50; x++; return x;}
int f3() { x *= 10; return x;}
int f4(int a, int b , int c , int d) {
return a+b+c+d;}
```

X = 10 = x = 10×10 = 100

26 51 100 51

1 + 26 + 51 + 100 + 51

= 229

229

Slide

Question

PYQ - 2024

Consider the following C program. Assume parameters to a function are evaluated from right to left.

```
#include <stdio.h>
int g(int p) { printf("%d", p); return p; }
int h(int q) { printf("%d", q); return q; }
void f(int x, int y) {
    g(x);
    h(y);
}
int main() {
    f(g(10),h(20));
}
```

Which one of the following options is the CORRECT output of the above C program?

2010 1020

(A) 20101020

(B) 10202010

(C) 20102010

(D) 10201020

1. g(10); 10
2. h(y);

f(g(10), h(20));

f(g(10), 20)

f(10, 20)

Slide

Question

#Q.

```
#include<stdio.h>
int bar(int y){
        int x = 20;
        x-=y;
        return x;
}
int foo(){
    int x = 1;
    x+=10;
    return bar(x);
}
int main(){
    int x,y;
    x = bar(15);
    y = bar(foo())+x;
    printf("%d\n", (x+y));
    return 0;
}
```

x = bar(15)

x = 5

y = bar(foo()) + 5

y = bar(bar(11)) + 5

y = bar(9) + 5

y = 11 + 5 = 16

16 + 5 = 21

Slide

The value printed by the program is

2 mins Summary

Topic — function

Topic — Activation Record

Topic — Activation Tree

Topic

Topic

Slide

THANK - YOU