Computer Science & IT

C Programming

Practice Classes

Lecture No. 01

By- Abhishek Sir

Recap of Previous Lecture

Topic String 2-D

Topic Structure

Topic

Topic

Topic

Slide

Topics to be Covered

Topic

practice problems

Topic

Topic

Topic

Topic

Slide

Question

oops

What is the value printed by the following C program?

```
#include < stdio.h >
 int f(int * a, int n){
        if (n <= 0)
                return 0;
        else if(*a % 2 = = 0)
                return * a + f(a +1, n -1);
        else
                return * a- f(a + l, n-1);
}
int main (){
        int a[] = {12,7,13,4,11,6};
        printf("%d", f(a,6));
        return 0;
}
```

100 104 108 112 116 120

(A) -9

(B) 5

(C) 15

(D) 19

f(a, 6) 13

12 + foo(104, 5) 3

↓

7 - foo(108, 4) 4

↓

13 - f(112, 3) 9

↓

4 + foo(116, 2) 5 5

↓

11 - foo(120, 1) 6 6

↓

6 + foo(124, 0) 0

Slide

Question

88. Consider the following program:

```
int f(int*p, int n) {
  if (n<=1)
          return 0;
  else
          return max(f(p+1,n-1), p[0] - p[1]);
}
int main() {100 104 108 112 116
  int a[ ]={3, 5, 2, 6 ,4};
  printf("%d", f(a, 5));
}
```

Note: max(x, y) *returns the maximum of* x *and* y.

The value printed by this program is________.

foo(100, 5) - $\underline{3}$

↓

max $\left(\overset{3}{f(104, 4)}, -2\right)$

↓

max $\left(\overset{2}{f(108, 3)}, 3\right)$

↓

max $\left(\overset{2}{f(112, 2)}, -4\right)$

↓

max $\left(\underset{0}{f(116, 1)}, 2\right)$

Slide

Question

What is printed by the following ANSI C program?

```
#include<stdio.h>
int main(int argc, char *argv[]){
int x = 1, z[2] = {10, 11};
   int *p = NULL;
   p = &x;
   *p = 10;
   p = &z[1];
   *(&z[0] + 1) += 3;
   printf("%d, %d, %d\n", x, z[0], z[1]);
   return 0;
}
```

(A) 1, 10, 11

(B) 1, 10, 14

(C) 10, 14, 11

(D) 10, 10, 14

Question

The most appropriate matching for the following pairs

Data structure

X: m=malloc(5);m= NULL;
Y: free(n); n->value=5;
Z: char *p; *p='a';

1: using dangling pointers
2: using uninitialized pointers
3. lost memory

is:

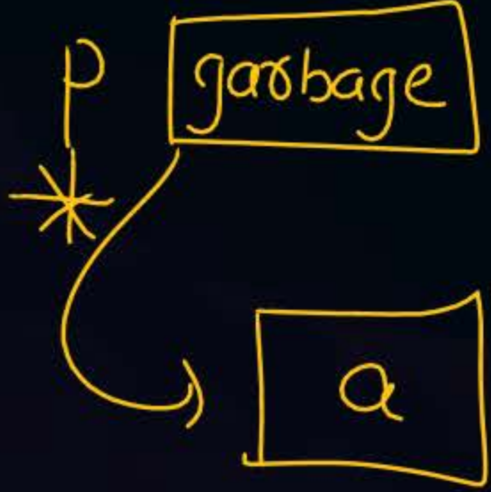
(a) X – 1 Y – 3 Z - 2

(b) X – 2 Y – 1 Z - 3

(c) X – 3 Y – 2 Z - 1

[✓ (d)] X – 3 Y – 1 Z - 2

char * p;



malloc - Memory allocation

m 1000 Address

free(1000) deallocate

1000

Slide

Question

Consider the following three C functions:

× $\boxed{10}$ ← deallocated

```
[P1] int*g(void){
        int x=10;
        return(&x);
     }
```

✓

```
[P2] int*g(void){
        int*px;
        *px=10;
        return px;
     }
```

← uninitializ

```
[P3] int*g(void){
        int* px;
        px =(int*)malloc(size of (int));
        *px=10;
        return px;
     }
```

Which of the above three functions are not likely to cause problems with pointers?

[✓ (a)] Only P3

(b) Only P1 and P3

(c) Only P1 and P2

(d) P1, P2 and P3

Slide

Question

"%%d"

Consider the following C-program: 14

```
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

Slide

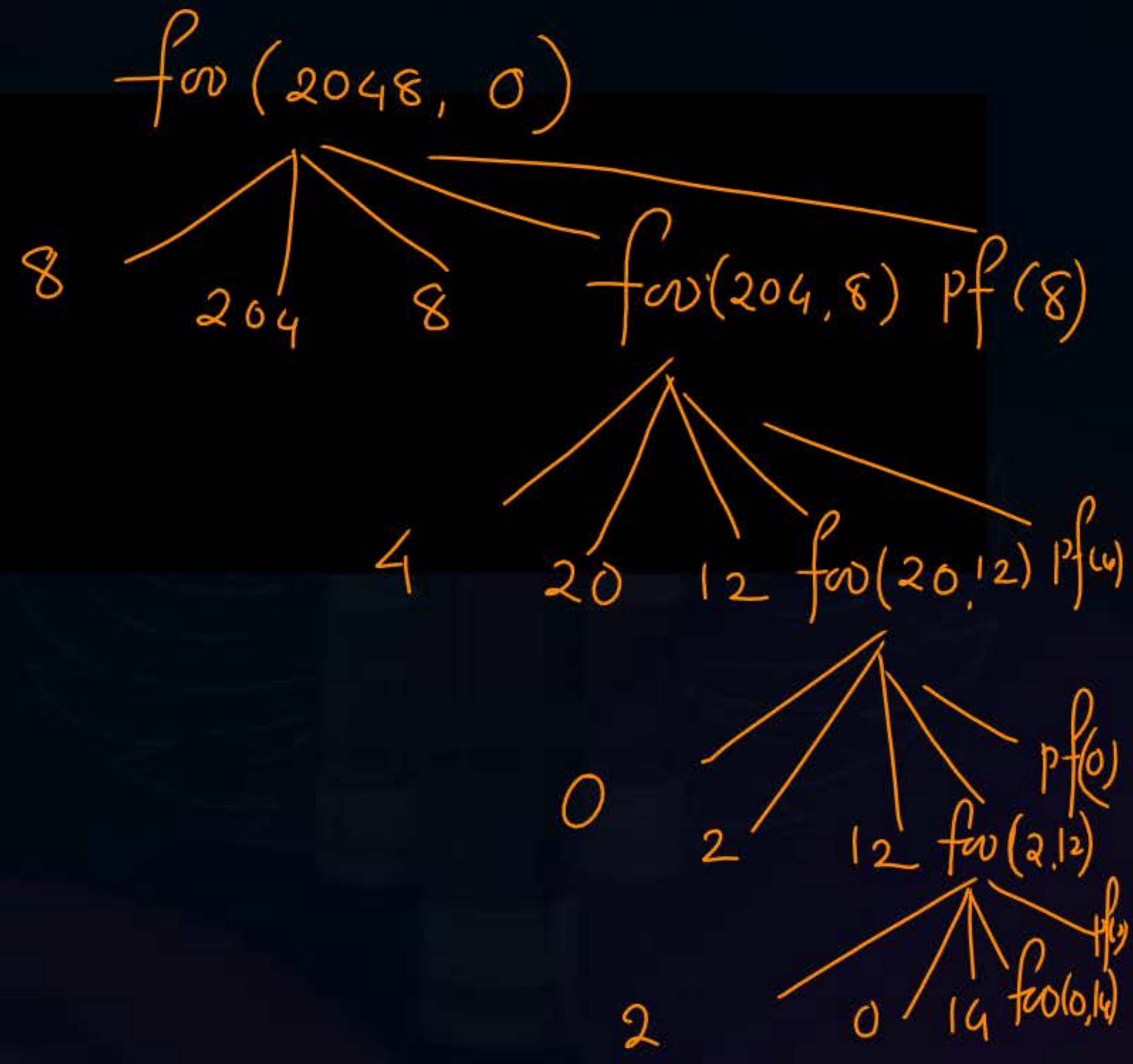
Question

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);
    }
```

output %d ;

a[i][j] = j[i[a]]; ✓



Slide

```
#include <stdio.h>
void swap (int *x, int *y)
{
   static int *temp;
   temp = x;
   x = y;
   y = temp;
}
```

```
void printab ()
{
   static int i, a = -3, b = -6;
   i = 0;   → Assignment
   while (i <= 4)
   {
      if ((i++)%2 == 1) continue;
      a = a + i;
      b = b + i;
   }
   swap (&a, &b);
   printf("a =  %d, b = %d\n", a,
b);
}
```

```
main()
{
   printab();
   printab();
}
```

(A) $a = 0, b = 3\ , a = 0, b = 3$

(B) $a = 3, b = 0,\ a = 12, b = 9$

(C) $a = 3, b = 6, a = 3, b = 6$

[✓ (D) $a = 6, b = 3\ , a = 15, b = 12$]

Slide

```
#include <stdio.h>
void swap (int *x, int *y)
{
    static int *temp;
    temp = x;
    x = y;
    y = temp;
}
```

Address swap

```
void printab ()
{
    static int i, a = -3, b = -6;
    i = 0;
    while (i <= 4)
    {
        if ((i++)%2 == 1) continue;
        a = a + i;
        b = b + i;
    }
    swap (&a, &b);
    printf("a =  %d, b = %d\n", a,
b);
}
```

← assignment

X

```
main()
{
    printab();
    printab();
}
```

a:-3 b=-6

i:0 -2 -5

i:1 ✓

i:2 Add3 1 -2

i:3 ✓

i:4 Add 6 3

5

Slide

Question

Consider the following C program:

```
#include<stdio.h>
void fun1(char *s1, char *s2){
        char *tmp;
        tmp = s1;
        s1 = s2;
        s2 = tmp;
}
void fun2(char **s1, char **s2){
        char *tmp;
        tmp = *s1;
        *s1 = *s2;
        *s2 = tmp;
}
```

```
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;
}
```

$\boxed{Hi}\boxed{\backslash 0}$ 100   Bye

str1 $\boxed{\cancel{100}\ 200}$ 300   str2 $\boxed{\cancel{200}\ 100}$ 400

S1 $\boxed{100}$ S2 $\boxed{200}$

S1 $\boxed{200}$ S1 $\boxed{100}$ *

S1 $\boxed{300}$ S2 $\boxed{400}$

temp : 100

Hi Bye

Bye Hi

[✓ (A) Hi Bye Bye Hi]

(B) Hi Bye Hi Bye

(C) Bye Hi Hi Bye

(D) Bye Hi Bye Hi

Slide

Question

Which one of the choices given below would be printed when the following program is executed?

```
#include<stdio.h>
struct tes{
 int i;
char *c;      100                    200
}st[5]={{5,"becomer"},{4,"better"},{6,"jungle"},
{8,"ancestor"},{7, "brother"}};     300
    400              500

main (){
 struct tes *p=st;
 p+=1;
 ++(p->c); //
 printf("%s, ",p++->c); ✓ etter
 printf("%c",*++p->c);
}
```

p++ → c

612 → c

p = | 600 | 612 | 624 |

++ (p → c)

(A) jungle, n

✓ (B) etter, u

(C) cetter, k

(D) etter, n

++ (200)

st[0]

5 4	100 8	4	~~200~~ 201	6	~~300~~ 301	8	400	7	500

600 612 624 636 648

Slide

Question

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;
      }
```

Slide The value printed by the program is __________

Question

Consider the following C program

```
#include <stdio.h>
int main(){
int a[] = {2, 4, 6, 8, 10};
int i, sum = 0, *ptr = a, **ptr1 = &ptr;
for (i = 0; i < 5; i++, ptr++ )
        sum = sum + (*ptr - i)*3 + (**ptr1 - i)*2;
printf("%d\n",sum);
return 0;
}
```

The output of the above C-program is _______.

100 104 108 112 116

Sum=0

ptr $\boxed{100}$
200

ptr1 $\boxed{200}$

i=0 0+6+4=10

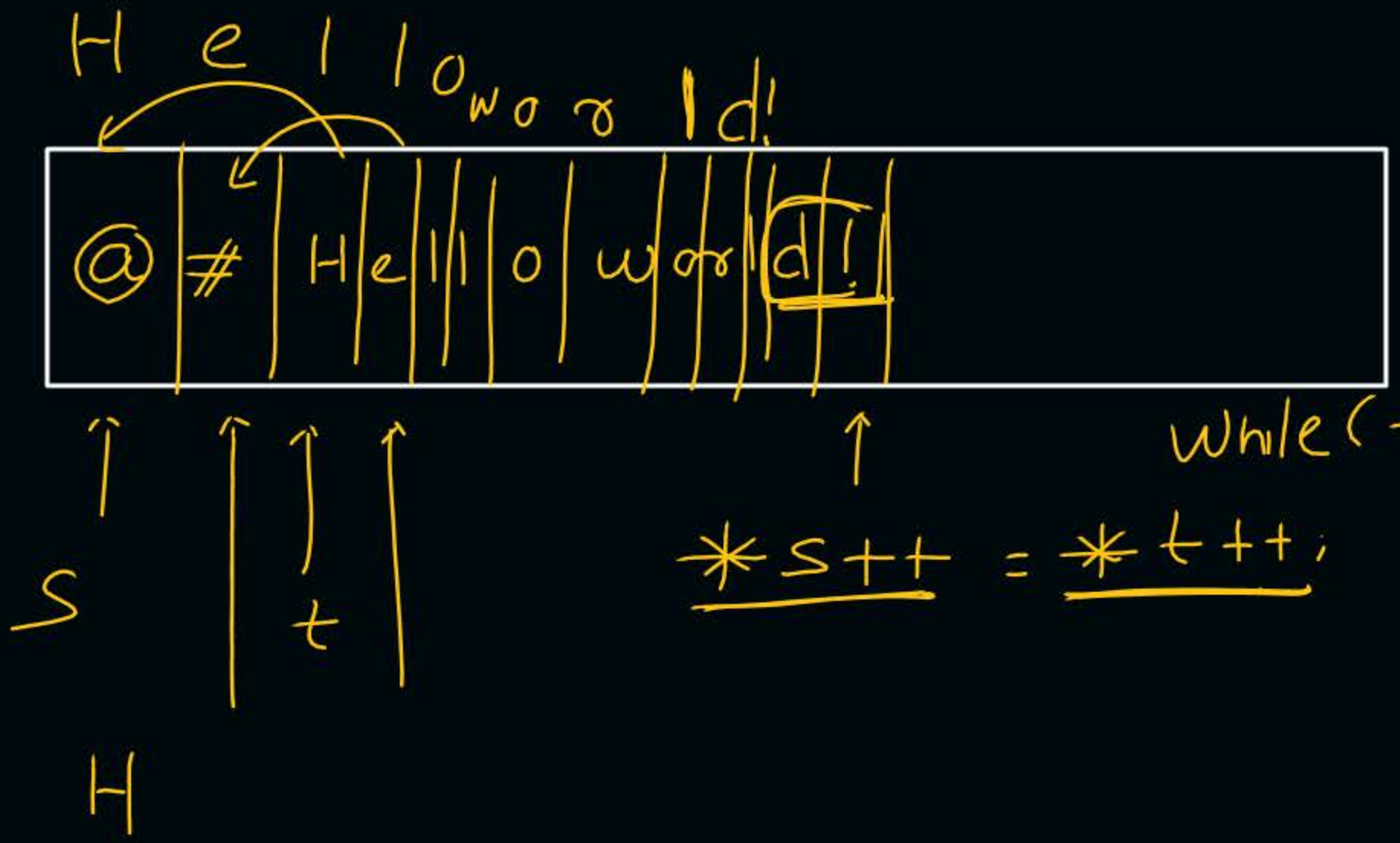
i=1 10+15=25

i=2 25+20=45

i=3 45+25=70

i=4 70+30=100

Slide



```
while(*t)
*s++ = *t++;
```

GATE 2020

C program is given below:

```
# include <stdio.h>
int main ()
{
        int i, j;
        char a [2] [3] = {{'a', 'b', 'c'}, {'d', 'e', 'f'}};
        char b [3] [2];
        char *p = *b;
        for (i = 0; i < 2; i++) {
              for (j = 0; j < 3; j++) {
              *(p + 2*j + i) = a [i] [j];
              }
        }
}
```

What should be the contents of the array b at the end of the program?

A. a b
c d
e f

B. a d
b e
c f

C. a c
e b
d f

D. a e
d c
b f

Slide

Question

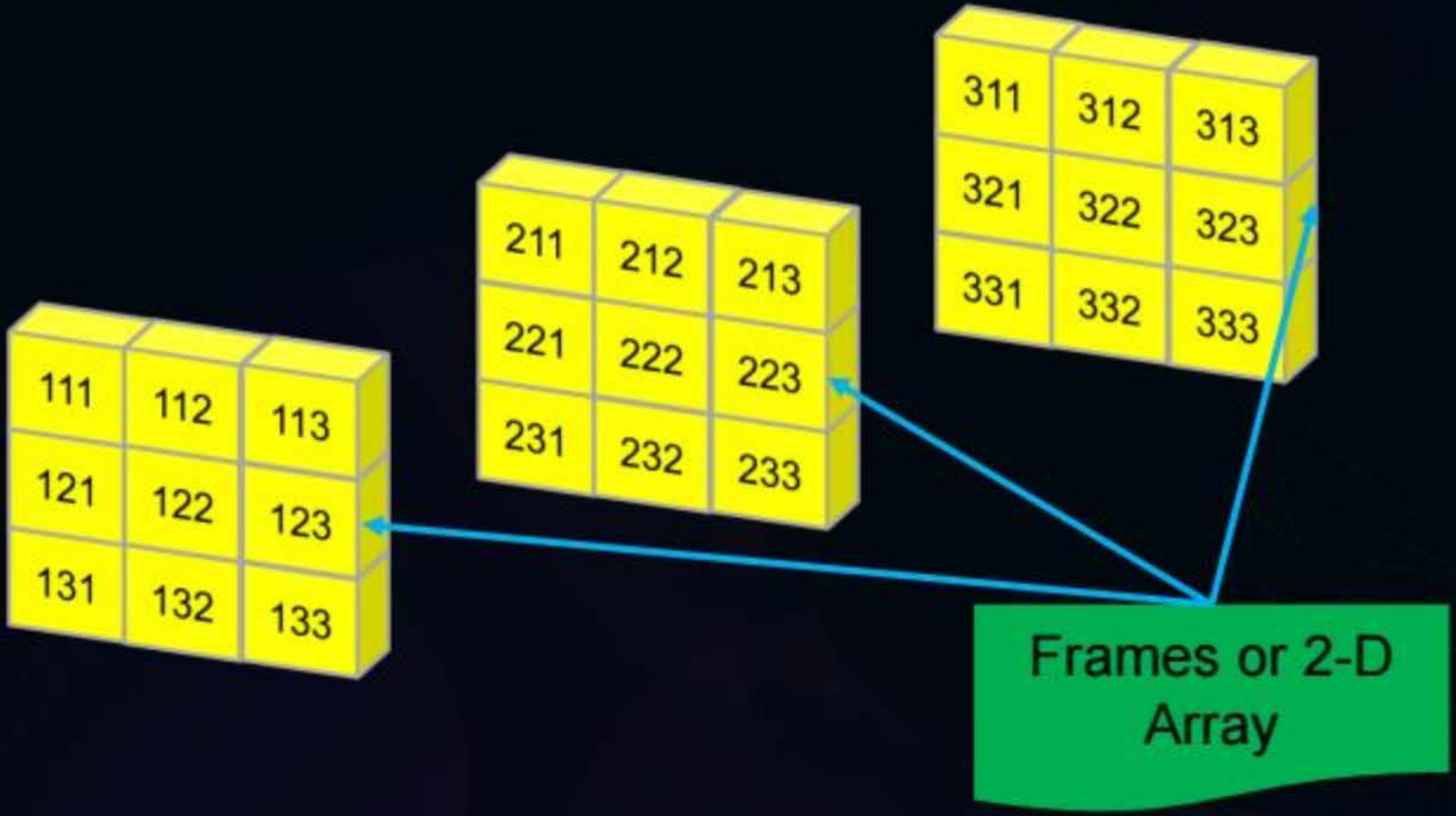
C program is given below:

```
# include <stdio.h>
int main ()
{
        int i, j;
        char a [2] [3] = {{'a', 'b', 'c'}, {'d', 'e', 'f'}};
        char b [3] [2];
        char *p = *b;
        for (i = 0; i < 2; i++) {
              for (j = 0; j < 3; j++) {
              *(p + 2*j + i) = a [i] [j];
              }
        }
}
```

What should be the contents of the array b at the end of the program?

Slide

3-D Array



111	112	113
121	122	123
131	132	133

211	212	213
221	222	223
231	232	233

311	312	313
321	322	323
331	332	333

Frames or 2-D Array

Question

What is printed by the following ANSI C program?

```
#include<stdio.h>
int main(int argc, char *argv[])
{
int a[3][3][3] = {{1, 2, 3, 4, 5, 6, 7, 8, 9},
                  {10, 11, 12, 13, 14, 15, 16, 17, 18},
                  19, 20, 21, 22, 23, 24, 25, 26, 27}};
int i = 0, j = 0, k = 0;
for( i = 0; i < 3; i++ ){
     for(k = 0; k < 3; k++ )
          printf("%d ", a[i][j][k]);
     printf("\n");
}
return 0;
}
```

(A) 1 2 3
10 11 12
19 20 21

(B) 1 4 7
10 13 16
19 22 25

(C) 1 2 3
4 5 6
7 8 9

(D) 1 2 3
13 14 15
25 26 27

Slide

Question

Consider the following ANSI C function

```
int SimpleFunction(int Y[], int n, int x) {
int total = Y[], loopIndex;
For (loopIndex = 1; loopIndex<=n-1; loopIndex++)
     total = x * total Y[loopIndex];
return total;
}
```

Let Z be an array of 10 elements with Z[i] = 1 for all I such that . The value returned by simpleFunction (Z, 10, 2) is ______.

Slide

Question

```
#include <stdio.h>
int al[] = {6, 7, 8, 18, 34, 67};
int a2[] = {23, 56, 28, 29};
int a3[] = {-12, 27, -31};
int *x[] = {al, a2, a3};
void print(int *a[]){
         printf("%d",a[0][2]);
}
int main(){
  print(x);
  return 0;

}
```

Output of the program is________

THANK - YOU