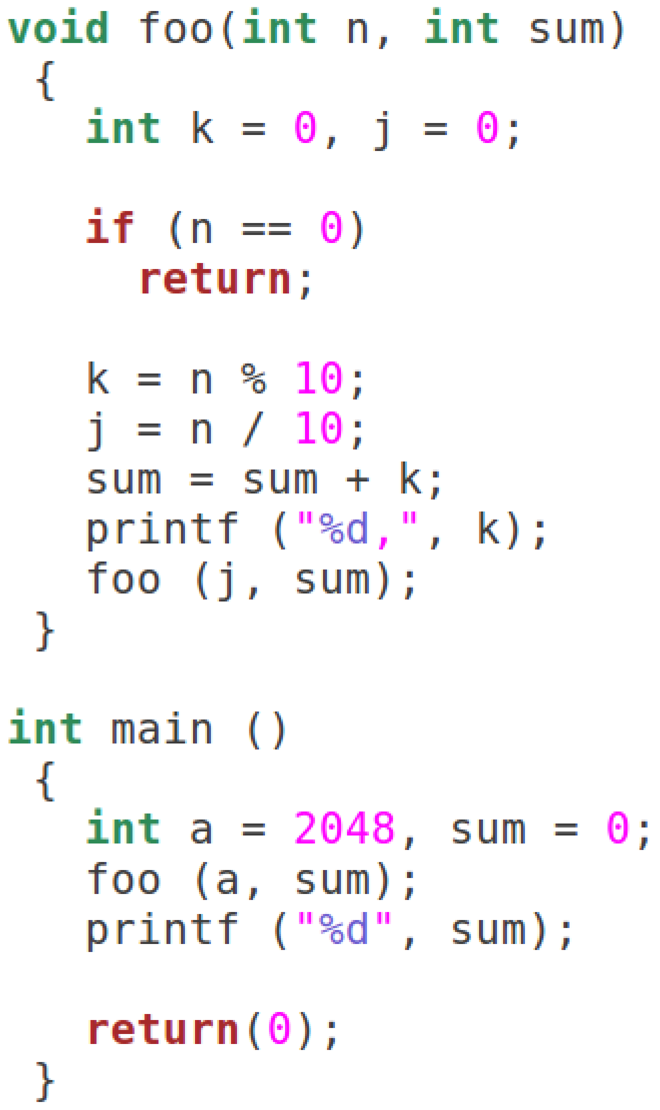
Introduction to C programming for Placements

1. **Consider the following C-program: What does the program print? (Eliminate wrong answers or draw recursion tree)**



```
void foo(int n, int sum)
 {
   int k = 0, j = 0;

   if (n == 0)
     return;

   k = n % 10;
   j = n / 10;
   sum = sum + k;
   printf ("%d,", k);
   foo (j, sum);
 }

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

   return(0);
 }
```

Mark only one oval.

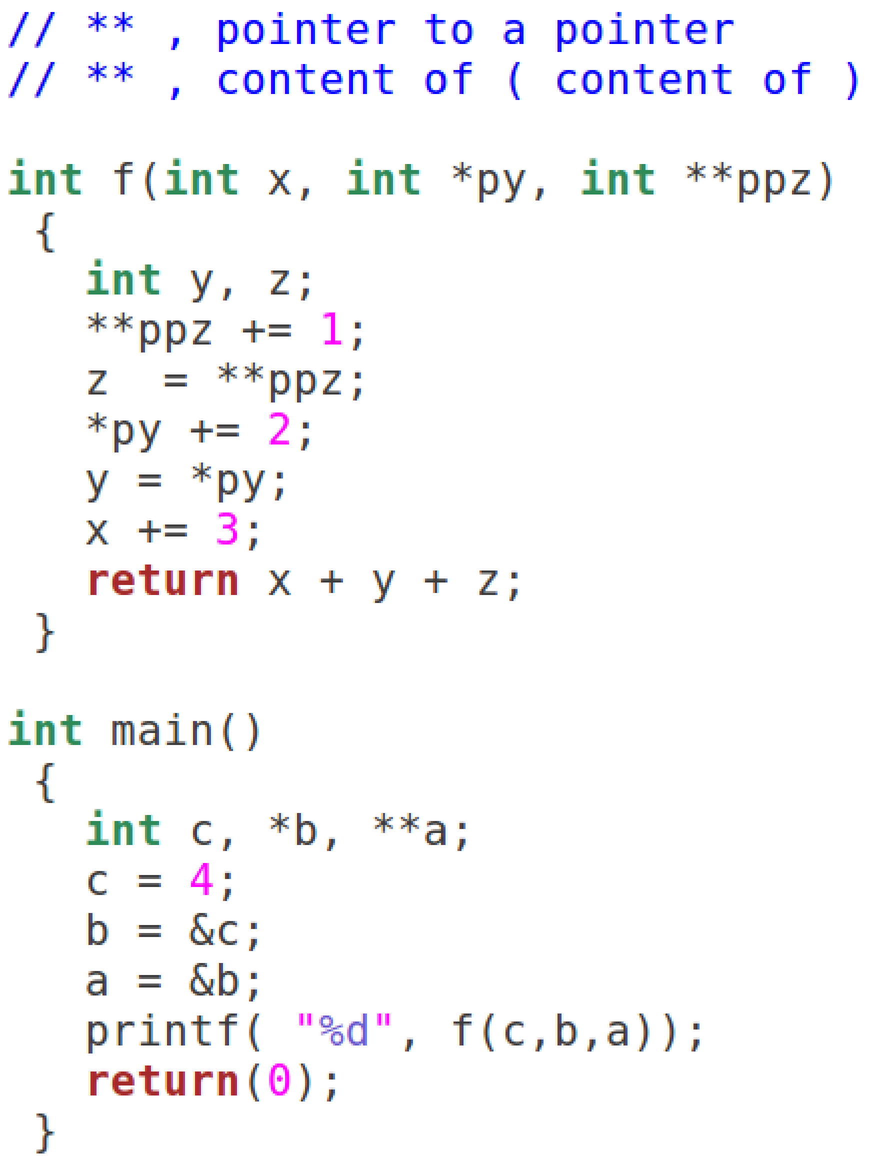
○ 2, 0, 4, 8, 0

○ 8, 4, 0, 2, 14

○ 2, 0, 4, 8, 14

○ 8, 4, 0, 2, 0

2. **What is printed by the following C program? (Draw diagram to get clear idea)**



```
// ** , pointer to a pointer
// ** , content of ( content of )

int f(int x, int *py, int **ppz)
 {
   int y, z;
   **ppz += 1;
   z  = **ppz;
   *py += 2;
   y = *py;
   x += 3;
   return x + y + z;
 }

int main()
 {
   int c, *b, **a;
   c = 4;
   b = &c;
   a = &b;
   printf( "%d", f(c,b,a));
   return(0);
 }
```

Mark only one oval.

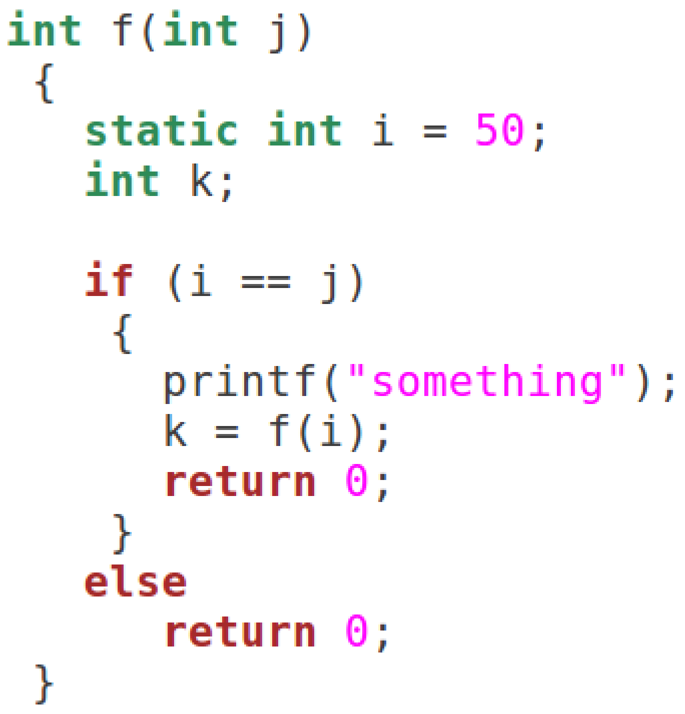
○ 21

○ 18

○ 22

○ 19

3. **Consider the C function given below. Which one of the following is TRUE? (Eliminate wrong answers or draw recursion tree)**



```
int f(int j)
 {
   static int i = 50;
   int k;

   if (i == j)
    {
      printf("something");
      k = f(i);
      return 0;
    }
   else
      return 0;
 }
```

Mark only one oval.

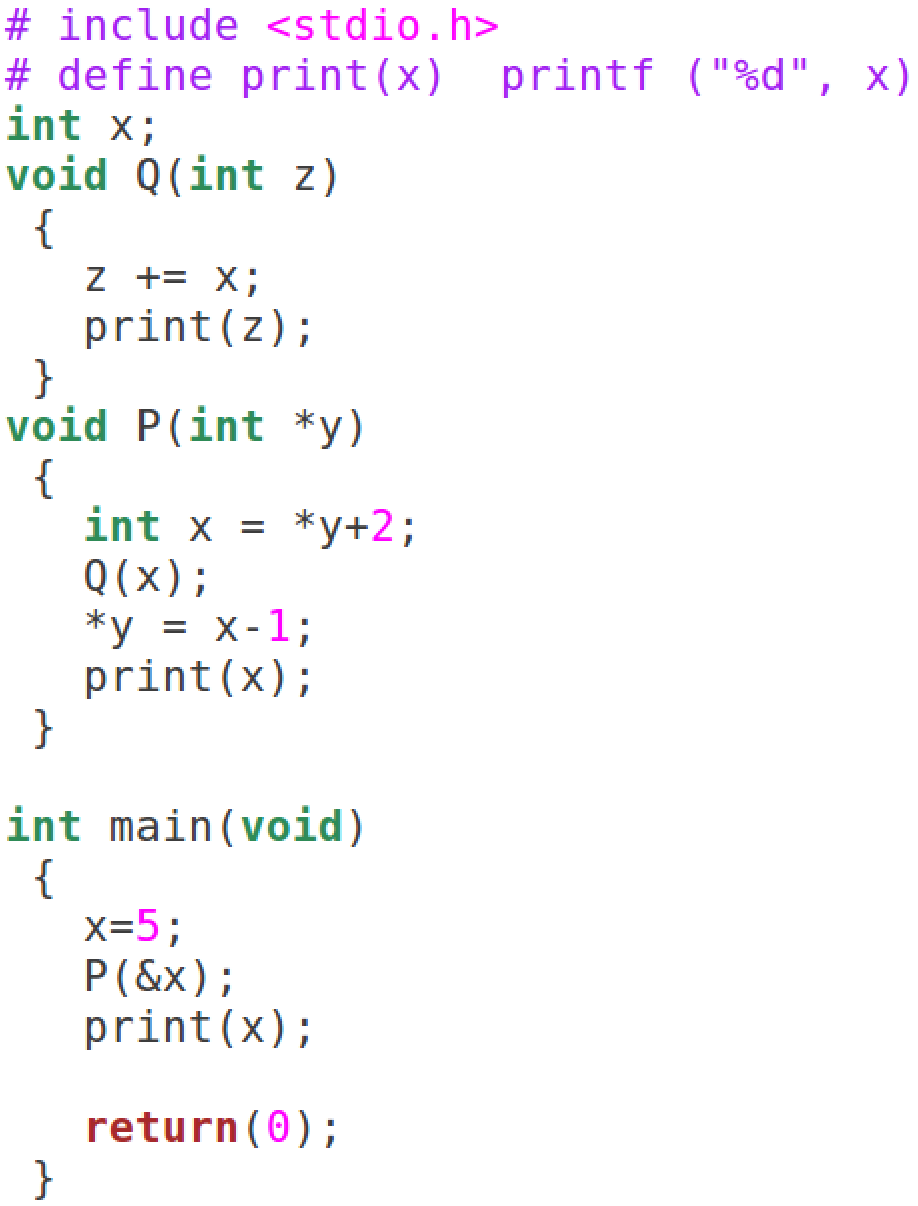
○ The function returns 0 for all values of j.

○ The function will exhaust the runtime stack or run into an infinite loop when j = 50

○ The function prints the string "something" for all values of j.

○ The function returns 0 when j = 50.

4. **Consider the C program shown below. The output of this program is : (Eliminate wrong answers)**



```
# 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);
 }

int main(void)
 {
   x=5;
   P(&x);
   print(x);

   return(0);
 }
```

Mark only one oval.

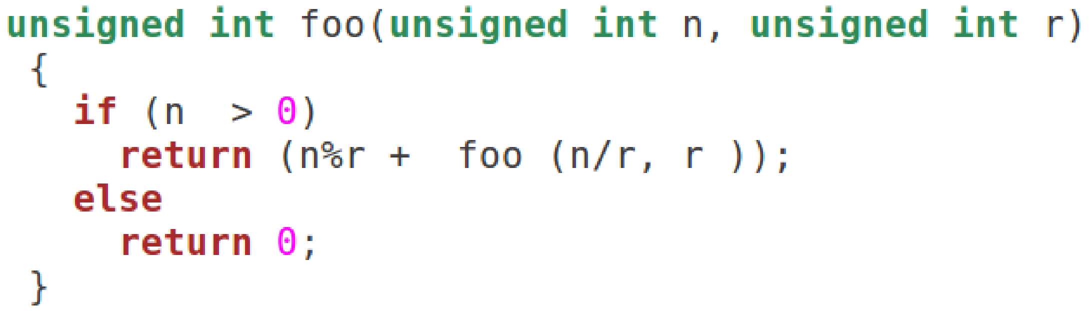
○ 1466

○ 1276

○ 22 12 11

○ 766

5. **What is the return value of the function foo when it is called as foo(345, 10) ? (Eliminate wrong answers or draw recursion tree and trace back)**



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

Mark only one oval.

○ 12

○ 140

○ 345

○ 10

Powered by Google Forms