Given a Java-like program as follows.

class A

{

public

int n;

public int f1()

{

n+=4; int m= f2();return n + m + 1;

}

public int f2()

{

n += 1;

return n;

}

public int f3()

{

f1();

return n;

}

};

class B : A

{

public int f1()

{

n-=4; int m= f2(); return n - m;

}

public int f2()

{

n += n;

return n;

}

};

}

…

A b = new B(); b.n = 4;

cout << b.f3(); //1

State the value displayed after the statement commented as //**1** executes in the following cases:

In static binding, object b holds A type, since b is initiallized by A type.

In dynamic binding, object b holds B type, since the pointer of B is initiallized by B type.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | f1 | f2 | | f3 | |
| Case | f1 | f2 | n1 | n2 | m | return | b.f3() = n |
| a | A | A | 8 | 9 | 9 | 19 | 9 |
| b | B | B | 0 | 0 | 0 | 0 | 0 |
| c | A | B | 8 | 16 | 16 | 33 | 16 |
| d | B | A | 0 | 0 (static -> unchanged) | 1 | 0 | **0** |

|  |  |  |
| --- | --- | --- |
| **Case** | **static binding** | **dynamic binding** |
| a | f1,f2 |  |
| b |  | f1,f2 |
| c | f1 | f2 |
| d | f2 | f1 |