CSDS 600 Programming Language Concepts: Scoping Written Exercise

Problem 1: Consider the following Java code (assuming Java allows methods to be declared inside other methods:

```
public class AClass {
  private static int a = 10;
  private static int b = 20;

  public static int bmethod(int y) {
    int b = 30;

    public static int dmethod(int z) {
       int a = z;
       return a + cmethod(z);
    }

    return a + b + dmethod(y);
  }

  public static int cmethod(int x) {
    int a = 40;
    if (x == 0) {
      return a + b;
    }
    else {
      return bmethod(x-1) + a + b;
    }
  }

  public static void main (String[] args) {
    int b = 2;
    System.out.println(cmethod(b) + a + b);
  }
}
```

What is the value printed by the `System.out.println` statement when the `main` method is run if:
a) Java uses static scoping
b) Java uses dynamic scoping
Be sure to trace your reasoning and justify your answer!

Answer:

a) For static scoping, we use the variable definition in the same block.

```
main:
  System.out.println(cmethod(2) + 10 + 2);

  cmethod (x=2):
    return bmethod(1) + 40 + 20
```

```
    bmethod (y=1):
      return 10 + 30 + dmethod(1)

      dmethod (z=1):
        return 1 + cmethod(1)

        cmethod (x=1):
          return bmethod(0) + 40 + 20

          bmethod (y=0):
            return 10 + 30 + dmethod(0)

            dmethod (z=0):
              return 0 + cmethod(0)

              cmethod (x=0);
                return 40 + 20

            dmethod (z=0):
              return 0 + 60

          bmethod (y=0):
            return 10 + 30 + 60

        cmethod (x=1):
          return 100 + 40 + 20

      dmethod (z=1):
        return 1 + 160

    bmethod (y=1):
      return 10 + 30 + 161

  cmethod (x=2):
    return 201 + 40 + 20

main:
  System.out.println(261 + 10 + 2);

273
```

b) For dymanic scoping, we use the call stack to find the bindings

```
a = 10
b = 20
main:
  b = 2
  System.out.println(cmethod(2) + 10 + 2);

  cmethod (x=2):
    a = 40
```

```
    return bmethod(1) + 40 + 2

    bmethod (y=1):
      b = 30
      return 40 + 30 + dmethod(1)

      dmethod (z=1):
        a = 1
        return 1 + cmethod(1)

        cmethod (x=1):
          a = 40
          return bmethod(0) + 40 + 30

          bmethod (y=0):
            b = 30
            return 40 + 30 + dmethod(0)

            dmethod (z=0):
              a = 0
              return 0 + cmethod(0)

              cmethod (x=0);
                a = 40
                return 40 + 30

            dmethod (z=0):
              return 0 + 70

          bmethod (y=0):
            return 40 + 30 + 70

        cmethod (x=1):
          return 140 + 40 + 30

      dmethod (z=1):
        return 1 + 210

    bmethod (y=1):
      return 40 + 30 + 211

  cmethod (x=2):
    return 281 + 40 + 2

main:
  System.out.println(323 + 10 + 2);
```

```
335
```