

Matthew Louque

Java Hyphen Puzzle

I found two solutions.

The first one is that you can change the “i” in the last argument of the for loop to an “n” which causes n to be decreased by 1 each loop. This causes for the iteration of $I < n$ to look like $I = 0, n = 20$. Which results in a true. N then becomes 19. This means that after 20 loops $I = 0$ and $n = 0$, which then stops the loop because I is no longer less than n.

```
public class Puzzle
{
    public static void main(String[] args)
    {
        int i, n = 20;

        for (i=0; i<n; n--)
            System.out.print("-");
        System.out.println();
    }
}
```

The second solution was less obvious. I added a “-” in front of the I in the middle argument of the for loop to have the conditional statement that is check at the beginning of each loop be $-I < n$. I is still being decreased by one each loop, which causes the conditional value to look like $-0 < 20, -(-1) < 20, -(-2) < 20 \dots -(-19) < 20, -(-20) < 20$.

```
public class Puzzle
{
    public static void main(String[] args)
    {
        int i, n = 20;

        for (i=0; -i<n; i--)
            System.out.print("-");
    }
}
```

```
System.out.println();
```

```
}
```

```
}
```