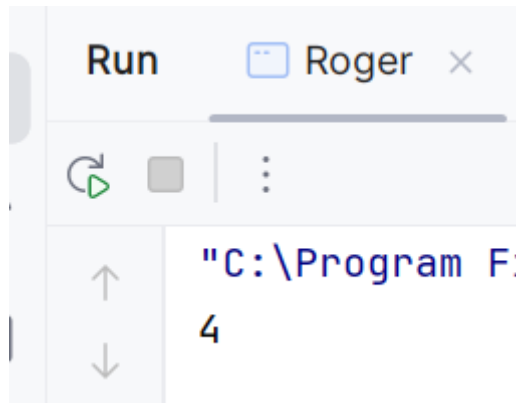


```

public class Roger {
    public static int roger(int k, int b) {
        if (b == 1) return 1;
        else if (k % b == 0) return 1 + roger(k, --b);
        else return 0 + roger(k, --b);
    }
    public static void main(String[] args) {
        System.out.println(roger(8, 8));
    }
}

```

Berapakah return value dari pemanggilan roger(8,8) ? 4



Penjelasan:

Diketahui:

- Jika $b == 1$, maka mengembalikan 1
- Jika $k \% b == 0$, maka mengembalikan $1 + \text{roger}(k, --b)$
- Selain itu, maka mengembalikan $0 + \text{roger}(k, --b)$

Jawab:

- $\text{roger}(8,8) \rightarrow 8 \neq 1 \parallel 8 \% 8 == 0$, maka mengembalikan $1 + \text{roger}(8,7)$
- $\text{roger}(8,7) \rightarrow 7 \neq 1 \ \&\& \ 8 \% 7 \neq 0$, maka mengembalikan $0 + \text{roger}(8,6)$
- $\text{roger}(8,6) \rightarrow 6 \neq 1 \ \&\& \ 8 \% 6 \neq 0$, maka mengembalikan $0 + \text{roger}(8,5)$
- $\text{roger}(8,5) \rightarrow 5 \neq 1 \ \&\& \ 8 \% 5 \neq 0$, maka mengembalikan $0 + \text{roger}(8,4)$
- $\text{roger}(8,4) \rightarrow 4 \neq 1 \parallel 8 \% 4 == 0$, maka mengembalikan $1 + \text{roger}(8,3)$
- $\text{roger}(8,3) \rightarrow 3 \neq 1 \ \&\& \ 8 \% 3 \neq 0$, maka mengembalikan $0 + \text{roger}(8,2)$
- $\text{roger}(8,2) \rightarrow 2 \neq 1 \ \&\& \ 8 \% 2 == 0$, maka mengembalikan $1 + \text{roger}(8,1)$
- $\text{roger}(8,1) \rightarrow 1 == 1$, maka mengembalikan 1

$$1 + 0 + 0 + 0 + 1 + 0 + 1 + 1 = 4$$

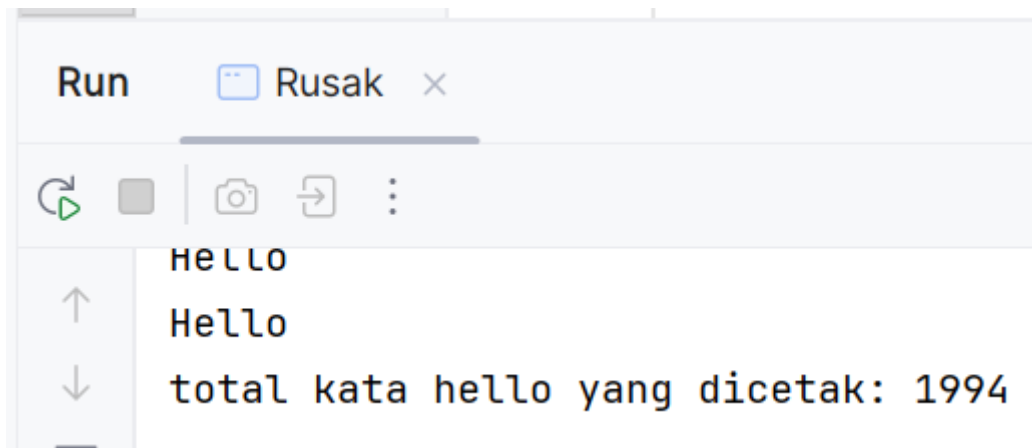
```

public class Rusak {
    static int helloCount = 0;
    public static void rusak (int n) {
        if (n == 0) {
            return;
        }
        for (int i = 0; i < n; i++) {
            System.out.println("Hello");
            helloCount++;
        }

        rusak((int)n/2);
    }
    public static void main(String[] args) {
        rusak(1000);
        System.out.println("total kata hello yang dicetak: " +
            helloCount);
    }
}

```

Berapa kali string “Hello” dicetak setelah pemanggilan rusak(1000)? 1994



Penjelasan:

Diketahui:

Method rusak akan menampilkan kata “Hello” sebanyak input n, method ini memanggil dirinya secara rekursif dan n akan dibagi 2 setiap kali sampai n bernilai 0.

Jawab:

- rusak(1000), menampilkan 1000 “Hello”
- rusak(1000/2) -> rusak(500), menampilkan 500 “Hello”
- rusak(500/2) -> rusak(250), menampilkan 250 “Hello”
- rusak(250/2) -> rusak(125), menampilkan 125 “Hello”
- rusak(125/2) -> rusak(62), menampilkan 62 “Hello”
- rusak(62/2) -> rusak(31), menampilkan 31 “Hello”
- rusak(31/2) -> rusak(15), menampilkan 15 “Hello”

- rusak(15/2) -> rusak(7), menampilkan 7 “Hello”
- rusak(7/2) -> rusak(3), menampilkan 3 “Hello”
- rusak(3/2) -> rusak(1), menampilkan 1 “Hello”
- rusak (1/2) -> rusak(0), n = 0 selesai.

$$1000 + 500 + 250 + 125 + 62 + 31 + 15 + 7 + 3 + 1 = 1994$$