

## ZV-08



Java nyelven!

Írjon olyan függvényt vagy metódust, amely egy természetes szám esetén kiírja, hogy a 9-es számjegyből hány darabot tartalmaz!

---

```
public static int countNine(int number) {  
    int count = 0;  
    while (number > 0) {  
        if (number % 10 == 9) {  
            count++;  
        }  
        number /= 10;  
    }  
    return count;  
}
```

Ez a függvény egy természetes számot vár paraméterként, és visszaadja, hogy hány darab 9-es számjegy van benne. A függvény while ciklussal végigmegy a számjegyeken, és ha talál egy 9-est, akkor növeli a számlálót.

---



Java nyelven!

Írjon olyan függvényt vagy metódust, amely egy természetes számról eldönti, hogy a kettes számrendszerbeli felírásában a jobbról második bitje 1 vagy 0!

---

```
public static boolean isSecondBitOne(int num) {  
    // Shift the number two bits to the right, then & with 1  
    return (num >> 1 & 1) == 1;  
}
```

A fenti kód egy metódust definiál, amely egy `int` típusú számot vár paraméterként, és visszaadja, hogy a kettes számrendszerbeli felírásában a jobbról második bitje 1 vagy 0. A `>>` operátorral a számot két bitet jobbra shifteljük, majd az `&` operátorral megvizsgáljuk, hogy az eredmény 1-e, ami azt jelenti, hogy a második bit 1. Ha az eredmény igaz, akkor a metódus `true`-t ad vissza, különben `false`.

---

Magyarázd el részletesebben!

---

A fenti kód egy `isSecondBitOne` nevű metódust definiál, amely egy `int` típusú számot vár paraméterként. A metódus visszaadja, hogy a kettes számrendszerbeli felírásában a jobbról második bitje 1 vagy 0.

Az első lépés, hogy a számot két bitet jobbra shifteljük a `>>` operátorral. Ennek hatására a jobbról második bitet az első bit helyére hozzuk, így az első bit a második helyen lesz.

A következő lépésben az `&` operátorral az első bitet kapjuk meg az első helyről, és megvizsgáljuk, hogy az 1-e. Ha az 1-e, akkor az eredmény 1 lesz, különben 0.

---