

Programmazione

Sessione di laboratorio

Lab07 - Istruzione while.

Integer square root

Esercizio

Scrivere un metodo chiamato `integerSquareRoot` che, dato in input un numero positivo n di tipo `int`, restituisca la sua radice quadrata intera x , ovvero il massimo valore intero x tale che $x \cdot x \leq n$.

Esempi

- $n = 1$: 1
- $n = 2$: 1
- $n = 15$: 3
- $n = 16$: 4

First last digit sum

Esercizio

Scrivere un metodo chiamato `firstLastDigitSum` che, dato in input un numero `n` di tipo `int` maggiore o uguale a 10, restituisca la somma della sua cifra più significativa e della sua cifra meno significativa.

Esempi

- `n = 10`: 1
- `n = 11`: 2
- `n = 23456`: 8
- `n = 95643`: 12

Invert number

Esercizio

Scrivere un metodo chiamato `invertNumber` che, dato in input un numero positivo `n` di tipo `int` la cui cifra meno significativa sia maggiore di 0, restituisca il numero intero ottenuto invertendo l'ordine delle sue cifre.

Esempi

- `n = 1`: 1
- `n = 11`: 11
- `n = 23456`: 65432
- `n = 95643`: 34659

Harmonic sum

Esercizio

Per ogni $k \geq 1$, il numero armonico $H(k)$ è definito come

$$H(k) = \sum_{i=1}^k \frac{1}{i}.$$

Scrivere un metodo chiamato `harmonicSum` che, dato in input un numero positivo x di tipo `int`, restituisca il più piccolo numero intero n tale che

$$H(1) + H(2) + \cdots + H(n) \geq x.$$

Esempi

- $x = 1$: 1
- $x = 2$: 2
- $x = 3$: 3
- $x = 5$: 4
- $x = 7$: 5
- $x = 1000$: 204

Duplicate digit

Esercizio

Scrivere un metodo chiamato `duplicateDigit` che, dato in input un numero positivo `n` di tipo `int`, restituisca il numero intero (di tipo `long`) ottenuto duplicando ogni sua cifra (ovvero scrivendo ogni sua cifra due volte).

Esempi

- `n = 1`: 11
- `n = 12`: 1122
- `n = 21`: 2211
- `n = 210`: 221100
- `n = 95043`: 9955004433