

INFORMATICA GENERALE

Homework 1: Algoritmi sui Numeri Naturali

docente: IVANO SALVO
Sapienza Università di Roma

pubblicazione: 29.III.2015 - consegna 12.IV.2015

Esercizio 1 Dato un intero strettamente positivo n , definiamo il *successivo* di un naturale $n > 1$ come segue:

$$\begin{array}{ll} n/2 & \text{se } n \text{ è pari} \\ 3n + 1 & \text{se } n \text{ è dispari } > 1 \end{array}$$

1 non ha successivi.

Scrivere un programma che letto in input un numero intero positivo n produca in output il numero di passi necessari affinché la sequenza che comincia con n raggiunga il numero 1.

ESEMPIO: La sequenza che comincia con 3, prosegue con 10, 5, 16, 8, 4, 2, 1: quindi in questo caso il programma dovrebbe stampare 7.

Esercizio 2 Scrivere un programma C che presi in input due interi positivi n e k ($n \geq 0$, $k \geq 1$) restituisca in output la *radice intera k -esima di n* . Più formalmente, il programma deve restituire in output un numero intero r , per cui sia verificata la disuguaglianza $r^k \leq n < (r + 1)^k$.

ESEMPIO: Presi in input 11 e 2, deve essere restituito come risultato 3, in quanto $3^2 = 9 \leq 11 < 4^2 = 16$.

Presi in input 27 e 3, deve essere restituito come risultato 3, in quanto $3^3 = 27 \leq 27 < 4^3 = 81$.

Esercizio 3 Un intero positivo n si dice *perfetto* se è uguale alla somma di tutti i suoi divisori propri (quindi incluso 1 ed escluso n). Ad esempio, $28 = 1+2+4+7+14$ è perfetto. n si dice *abbondante* se la somma di tutti i suoi divisori propri è maggiore di n . Ad esempio $24 < 1+2+3+4+6+8+12 = 36$ è abbondante. n si dice *difettivo* se la somma di tutti i suoi divisori propri è minore di n . Ad esempio $27 > 1+3+9 = 13$.

Scrivere una programma C che preso in input un numero positivo n stampi 0 se n è un numero perfetto, -1 se è difettivo ed 1 se è abbondante.