

Algoritmo per la riduzione di una frazione (ai minimi termini)

1) Una frazione è rappresentata da due numeri interi. Per rappresentare gli stessi è possibile utilizzare una opportuna struttura definita da due interi.

```
struct frazione {  
    int numeratore;  
    int denominatore;  
};
```

2) Siccome non tutte le frazioni sono già in forma ridotta (ad esempio $20 / 15$), può essere opportuno o richiesto effettuarne la riduzione (ad esempio $20 / 15 = 4 / 3$). La forma ridotta è una forma non più riducibile ulteriormente, vale a dire ove numeratore e denominatore non sono più divisibili, per alcun fattore comune. E' importante notare che il valore della frazione non cambia.

3) Per ridurre la frazione si inizia a dividerla per divisori più bassi possibili, vale a dire si inizia dal valore 2¹.

4) Se entrambi numeratore e denominatore sono divisibili per un certo divisore, si effettua, come ovvio, l'effettiva divisione delle due parti della frazione per il divisore attualmente considerato.

5) Quando invece il numeratore, il denominatore od entrambi siano non divisibili per il divisore candidato, è necessario passare ad un successivo divisore. Un modo per passare a divisori successivi è quello di passare al numero successivo. In effetti tale sistema non è efficiente in quanto aver controllato la divisibilità per 2 comporta ad esempio di aver controllato quella per 4 (= 2×2), per 8 (= $2 \times 2 \times 2$), per 16 e così via. Da questa riflessione si deduce che i divisori dovranno essere i soli numeri primi. Più corretto è quindi passare, dato il corrente divisore primo, a cercare il successivo numero primo, per verificare se divide numeratore e denominatore.

6) Questo procedimento di test e ricerca dei divisori deve chiaramente avere una condizione di stop². Essa risulta verificato quando numeratore e denominatore non sono sicuramente multipli del corrente divisore primo ed anche dei successivi, e questo avviene quando il corrente divisore primo e i suoi successivi, sono superiori sia al denominatore che al numeratore.

7) In questo caso (6) il processo di riduzione si ferma e viene stampata la frazione ridotta.

¹ Il valore 1 divide sempre e comunque qualunque numero lasciandolo invariato e quindi non è un divisore "utile".

² In una elaborazione reale nessun procedimento può continuare all' infinito.

Algoritmo di riduzione di una frazione

