

Il programma, lette 2 frazioni (cioè 4 interi positivi: numeratore e denominatore della prima frazione, numeratore e denominatore della seconda frazione), espone a video la frazione risultato, dopo averla semplificata (se necessario).

I passi da effettuare sono i seguenti:

- lettura delle 2 frazioni (num1, den1, num2, den2)
- calcolo del denominatore **den** della frazione risultato (cioè il m.c.m. dei denominatori)
- calcolo del numeratore **num** della frazione risultato
- semplificazione della frazione risultato (num e den vanno divisi per il proprio M.C.D. se maggiore di 1)

- scrittura a video delle frazioni da sommare (input) e della frazione risultato.

Esempi: $1/5 + 2/5 = 3/5$

$$7/5 + 3/5 = 10/5 = 2$$

Il m.c.m. viene calcolato utilizzando una funzione di nome **micomu** (algoritmo descritto in basso) che prevede in input 2 parametri interi **alfa** e **beta** e restituisce il valore finale calcolato per la variabile **ma**.

Il M.C.D. è calcolato utilizzando una funzione di nome **macodi** (algoritmo descritto in basso) che prevede in input 2 parametri interi positivi **uno** e **due** e restituisce il valore finale calcolato per la var. **num**.

rappresentazione algoritmo in PSEUDOCODIFICA

Funzione principale

```

INIZIO
leggiFrazioni()
scriviFrazioniInput()

den ← micomu (den1, den2)
num ← (den / den1) * num1
      + (den / den2) * num2

//-- scrive la frazione risultato
scriviFraz (num, den)

max ← macodi (num, den)
SE max <> 1
ALLORA
    num ← num / max
    den ← den / max
    SCRIVI (" = ")
    scriviFraz (num, den)
FINE SE
FINE
    
```

Funzione leggiFrazioni()

NON restituisce valori, NON ha parametri in ingresso

```

INIZIO
SCRIVI ("--- somma di 2 frazioni ---")
SCRIVI ("num. e denom. 1^ frazione: ")
LEGGI (num1, den1)
SCRIVI ("num. e denom. 2^ frazione: ")
LEGGI (num2, den2)
FINE
    
```

Funzione scriviFrazioniInput()

NON restituisce valori, NON ha parametri in ingresso

```

INIZIO
scriviFraz (num1, den1)
SCRIVI (" + ")
scriviFraz (num2, den2)
SCRIVI (" = ")
FINE
    
```

Funzione scriviFraz (Val nu, Val de)

NON restituisce valori, ha 2 parametri in ingresso di tipo intero passati per valore

```

INIZIO
SCRIVI (nu)
SE de <> 1 ALLORA
    SCRIVI (" / ", de)
FINE SE
FINE
    
```

Funzione micomu (Val alfa, Val beta)

Restituisce il m.c.m. (ma) di alfa e beta, quindi la funzione è di tipo intero ha 2 parametri in ingresso (alfa e beta) di tipo intero

```

INIZIO
ma ← alfa
mb ← beta
ESEGUI MENTRE (ma <> mb)
    SE ma > mb ALLORA
        mb ← mb+beta
    ALTRIMENTI
        ma ← ma+alfa
FINE SE
RIPETI
RITORNO-RESTITUISCI ma
FINE
    
```

Funzione macodi (Val uno, Val due)

Restituisce il M.C.D. (num) di uno e due, quindi la funzione è di tipo intero ha 2 parametri in ingresso (uno e due) di tipo intero

```

INIZIO
num ← uno
dvs ← due
ESEGUI
    res ← RESTO(num, dvs)
    num ← dvs
    dvs ← res
RIPETI MENTRE (dvs <> 0)
RITORNO-RESTITUISCI num
FINE
    
```

(NOTA: in C++ l'operatore % calcola il RESTO)

Funzioni: minimo comune multiplo (m.c.m.) e Massimo Comun Divisore (M.C.D.)

codifica in C++ (programma: **somma2frazioni.cpp**)

pag 2 di 2

prof.ssa P.Grandillo

```
#include <iostream>
using namespace std;
#define PIU '+'
#define EQ '='
#define DIV '/'
#define SP ' '
int i, k, num, den;
int num1, den1, num2, den2;
```

```
int micomu (int alfa, int beta){
    int ma, mb;
    ma = alfa;
    mb = beta;
    while (ma != mb)
    { if (ma > mb)
        mb = mb + beta;
      else
        ma = ma + alfa;
    }
    return ma;
}
```

```
int macodi (int uno, int due){
    int num, dvs, res;
    num = uno;
    dvs = due;
    do
    { res = num % dvs;
      num = dvs;
      dvs = res;
    } while (dvs != 0);
    return num;
}
```

```
void scriviFraz (int nu, int de) {
    cout<< nu;
    if (de == 1)//--se denominatore vale 1
        return; //--- uscita anticipata dalla funzione
    cout<< DIV;
    cout<< de; //-- scrittura denominatore
}
```

```
void scriviFrazioniInput (void){
    cout<< endl;
    scriviFraz (num1, den1);
    cout<< SP << PIU << SP;
    scriviFraz (num2, den2);
    cout<< SP << EQ << SP;
}
```

```
void leggiFrazioni (void) {
    cout<< endl;
    cout<< "--- somma di 2 frazioni ---\n";
    cout<< "\nnum. e denom. 1^ frazione: ";
    cin>>num1;
    do
        cin>>den1;
    while (den1 == 0);
    cout<< "\nnum. e denom. 2^ frazione: ";
    cin>>num2;
    do
        cin>>den2;
    while (den2 == 0);
}
```

```
int main(){
    int max;
    leggiFrazioni();
    scriviFrazioniInput();

    den = micomu (den1, den2);
    num = (den / den1) * num1
          + (den / den2) * num2;
    scriviFraz (num, den);

    max = macodi (num, den);
    if (max != 1) {
        num = num / max;
        den = den / max;
        cout<< SP << EQ << SP;
        scriviFraz (num, den);
    }
    cout<< endl << endl;
    system("pause");
}
```

```
>>somma2frazioni
--- somma di 2 frazioni ---
num. e denom. 1^ frazione: 1 6
num. e denom. 2^ frazione: 5 6
1/6 + 5/6 = 6/6 = 1
Premere un tasto per continuare . . . .
```

al richiamo della funzione **micomu** :

il parametro attuale **den1** viene copiato nel parametro formale **alfa** ,

il parametro attuale **den2** viene copiato nel parametro formale **beta** ;

al termine della esecuzione della funzione il parametro formale **ma** viene restituito e assegnato a **den**