

MODIFICA 3 per esercitazione 5

Scaricate la vostra esercitazione da gitlab. Create subito un **branch su GitLab** con il vostro nome e cognome (ad es. michele-rossi) e fatene il checkout sulla vostra macchina. Verificate di star lavorando sul branch appena creato (questo è il branch su cui andrete a fare il commit alla fine).

Non utilizzate la Continuous Integration per il vostro testing. Lavorate senza. Il commit va fatto una sola volta alla fine ed in presenza del docente.

Il vostro pc deve eseguire solo l'IDE (una sola istanza) e googleMeet ridotto ad icona dopo aver stabilito la connessione e condiviso l'intero schermo. Coloro che non rispetteranno le regole verranno esclusi dall'esame.

Si modifichi il compilatore **NewLang** per aggiungervi l'istruzione **mapsum** che permetta la chiamata di una funzione su più gruppi di argomenti e ne fa la somma.

Esempio valido:

```
integer a;
var b << 1, c << 2;

fun myfun(integer i, integer s): integer
{   return i*s;}

main
# istruzione mapsum
a << mapsum(myfun, (b+1, c*4+2), (2, 4), (4, 5)); # <=====
```

Nella traduzione in C l'istruzione **mapsum** diventa

```
a = myfun(b+1, c*4+2) + myfun(2, 4) + myfun(4, 5);    # <=====
```

Regole:

- il primo argomento dell'istruzione **mapsum** deve essere una funzione (come myfun) il cui tipo di ritorno è integer o real. I rimanenti argomenti sono gruppi fra parentesi con espressioni.
- il numero di espressioni in un gruppo fra parentesi ed i loro tipi devono essere gli stessi degli argomenti della funzione (ad es. 2 e di tipo intero per myfun)
- i gruppi in parentesi possono essere solo 3 o, nella versione avanzata, un numero $n \geq 3$

Casi di errori da considerare:

```
a << mapsum(myfun, (b+1, c*4+2), (2, 4.2), (4, 5))    # ERROR "argument type mismatch" (4.2 è un real e myfun accetta solo interi)
```

```
a << mapsum(myfun, (b+1), (2, 4.2), (4, 5))    # ERROR "wrong number of arguments"
```

```
string y;
y << mapsum(myfun, (b+1, c), (2, 4.2), (4, 5))    # ERROR "return value type mismatch" (y deve essere integer or real)
```

```
float x;
```

```
x <- mapsum(myfun, (b+1, c), (2, 4.2), (4, 5)) # ERROR "return value type mismatch" (myfun restituisce un intero ed x deve essere intero)
```

```
a <- mapsum(myfun, (b+1, c), (2, 4)) #SYNTAX ERROR (ci vogliono (almeno) tre gruppi di argomenti)
```

```
a <- mapsum(a, (b+1, c), (2, 4)) # ERROR "function name missing"
```