

Laboratorio di Programmazione

Lezione 10

3 febbraio 2004

Es. 1 Si scriva una funzione **eval-math-expr** che prenda in input una lista rappresentante un'espressione matematica (es. `'(+ (* 7 3) 5)`) e restituisca il valore dell'espressione stessa senza utilizzare la procedura **eval**.

Esempio

```
(eval-math-expr '(+ (* 7 3) 5))
```

```
-> 26
```

Es. 2 Si scriva una funzione **put-brackets** che prenda in input una lista di numeri e raggruppi a due a due i numeri “mettendo le parentesi”. Ogni coppia di numeri così ottenuta venga, inoltre, ordinata in modo crescente (il numero più piccolo in testa). Se il numero degli elementi della lista in input è dispari, l'ultima sottolista dell'output deve contenere solo un numero.

Esempio

```
(put-brackets '(100 7 23 15 11 30 55))
```

```
-> ((7 100) (15 23) (11 30) (55))
```

Es. 3 Si scriva una funzione **merge** che prenda in input due liste ordinate di numeri e restituisca una lista, anche essa ordinata, che contiene i numeri di entrambe.

Esempio

```
(merge '(1 5 100 121 150) '(2 3 4 5 6 111 121))
```

```
-> (1 2 3 4 5 5 6 100 111 121 121 150)
```

Es. 4 Usando le funzioni **put-brackets** e **merge**, si scriva una funzione, **simil-merge-sort**, che ordina una lista arbitraria di numeri.

Esempio

```
(simil-merge-sort '(1 160 30 2 42 51 12 1 90 93 21 25 11 7))
```

```
-> (1 1 2 7 11 12 21 25 30 42 51 90 93 160)
```