

```
computePar(int stuff)
```

```
    DP ← new int[1...n][1...n]
```

```
    last ← new int[1...n][1...n]
```

```
    // riempi diagonale principale
```

```
    da i ← 1 fino a n fai
```

```
    |   DP[i][i] ← 0
```

```
    // Tutta la logica dell'algoritmo
```

```
    da h ← 2 fino a n fai
```

```
    // h:  indice diagonale
```

```
        da i ← 1 fino a n - h + 1 fai
```

```
            // i:  riga
```

```
            int j ← i + h - 1
```

```
            // j:  colonna
```

```
            DP[i][j] ←  $+\infty$ 
```

```
            da k ← i fino a j - 1 fai
```

```
                int temp ← DP[i][k] + DP[k + 1][j] +  $c_{i-1} \cdot c_k \cdot c_j$ 
```

```
                se temp < DP[i][j] allora
```

```
                    // aggiorna l'ultimo prodotto
```

```
                    DP[i][j] ← temp
```

```
                    last[i][j] ← k
```