

Si scriva un programma completo C++ che, letta da input una sequenza di caratteri terminata dal carattere “.” (PUNTO), e rappresentante una espressione algebrica verifichi le seguenti condizioni:

1. che l'espressione sia “ben parentesizzata”
2. che non ci siano coppie di parentesi superflue.

Una espressione è “ben parentesizzata” se non ci sono parentesi “spaiate”, cioè prive della corrispettiva (parentesi aperte senza nessuna che le “chiuda”, né chiuse senza una che le abbia “aperte”). Es.: $(c + -(a +) - d)$ è una espressione ben parentesizzata. Invece $(c + (d- f)$ non lo è (infatti la prima parentesi rimane “aperta”).

Una coppia di parentesi è “superflua” se non contiene nulla al proprio interno. Es.: $(g + (f - () d))$ pur essendo ben parentesizzata contiene una coppia di parentesi superflue.

Il programma dovrà segnalare l'esito della verifica richiesta al punto 1) utilizzando, a seconda dei casi, una delle due istruzioni di stampa seguenti

```
cout << " ok.. l'espressione e' ben parentesizzata... " << endl;
cout << " attento! parentesi non bilanciate! " << endl;
```

Invece per segnalare l'esito della verifica richiesta al punto 2) il programma dovrà utilizzare, a seconda dei casi, una delle due istruzioni di stampa seguenti

```
cout << " ok.. l'espressione non contiene coppie di parentesi superflue " << endl;
cout << " attento! esistono coppie di parentesi superflue! " << endl;
```

ATTENZIONE: Si noti che affinché il programma possa essere gestito dal valutatore automatico, non dovrà contenere altre istruzioni di stampa oltre a quelle indicate sopra. **Esempio:** se l'espressione data in input fosse $(g + (f - () d))$ allora l'output del programma dovrebbe essere

```
ok.. l'espressione e' ben parentesizzata...
attento! esistono coppie di parentesi superflue!
```

Si noti che la sequenza introdotta da input è terminata dal carattere “.” (PUNTO). I caratteri che la compongono (escluso il tappo) costituiscono l'espressione sulla quale effettuare la verifica, e che negli esempi si parla direttamente di espressione (quindi ignorando il tappo).