

Esercizio – analizzatore identificatori del C++

Si consideri il linguaggio degli identificatori del C++ descritto dalla seguente grammatica BNF:

```
<id> ::= <lett> <id_resto>
<id_resto> ::= <lett> <id_resto> | <cifra> <id_resto> | ε
<lett> ::= a | b | ... | z | A | B | ... | Z | _ (underscore)
<cifra> ::= 0 | 1 | ... | 9
```

dove ε rappresenta la stringa vuota.

(a) Realizzare una funzione booleana di nome `id` che preso come suo parametro una stringa `s` restituisce come suo risultato `true` se la stringa appartiene al linguaggio generato a partire da `<id>`, `false` altrimenti. La funzione `id` utilizza (obbligatoriamente) una funzione booleana `id_resto` che, preso come suo parametro una stringa `r`, restituisce come suo risultato `true` se `r` appartiene al linguaggio generato a partire da `<id_resto>`, `false` altrimenti. La funzione `id_resto` deve essere obbligatoriamente definita in modo ricorsivo. SUGG. Passare come parametro della funzione `id_resto`, oltre alla stringa `s`, anche l'indice da cui iniziare l'analisi della stringa ...

(b) Scrivere un programma principale che legge da std input una stringa `s`, terminata da a capo (lunghezza max. 100 caratteri), e quindi determina, utilizzando (obbligatoriamente) la funzione `id` realizzata al punto (a), se `s` rappresenta o no un identificatore corretto.

N.B. Utilizzare soltanto stringhe “tipo C”. Non utilizzare variabili globali.

ESEMPI ESECUZIONE

Inserisci una stringa terminata da a capo:

-a

Errore

Inserisci una stringa terminata da a capo:

alb2

Identificatore corretto

Inserisci una stringa terminata da a capo:

lab

Errore

Inserisci una stringa terminata da a capo:

a

Identificatore corretto

Inserisci una stringa terminata da a capo:

1

Errore

Inserisci una stringa terminata da a capo:

_12

Identificatore corretto