

```

for statement
1
2
3
4
5
6

```

Παρατηρήσεις:

- Αν σε ένα βρόχο **for-to** (**for-downto**) οι ακραίες τιμές (κάτω και άνω φράγμα) της μεταβλητής ελέγχου είναι ίσες, τότε το σώμα του βρόχου εκτελείται μια φορά.
- Αν σε ένα **for-to** (**for-downto**) βρόχο ισχύει: άνω φράγμα < κάτω φράγμα (κάτω φράγμα < άνω φράγμα, αντίστοιχα), τότε το σώμα του βρόχου δεν εκτελείται καθόλου.
- Η μεταβλητή ελέγχου δεν είναι ορισμένη (δεν έχει τιμή) μετά το πέρας της εκτέλεσης του βρόχου.
- Μην αλλάζετε (με ανάθεση) την τιμή της μεταβλητής ελέγχου μέσα στο σώμα του βρόχου.
- Ακόμα και εάν αλλάξουν, μέσα στο σώμα του βρόχου, οι τιμές μεταβλητών που εμπλέκονται στις παραστάσεις των ακραίων τιμών του βρόχου, δεν επιφέρεται μεταβολή στον αριθμό των επαναλήψεων, διότι οι ακραίες τιμές προϋπολογίζονται.

```

• program star_banner(input, output);
  var i, j : integer;
begin
  for i:=1 to 5 do
    begin
      for j:=1 to 10 do write (' *');
      writeln
    end
  end.

```

Αποτέλεσμα:

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

Το προηγούμενο παράδειγμα εξηγεί την τεχνική των φωλιασμένων βρόχων (nested loops).

Σε αυτή τη σύνταξη των φωλιασμένων βρόχων παρατηρούμε ότι υπάρχει το εξωτερικό **for** loop (**for i:=1 to 5**) και το εσωτερικό (**for j:=1 to 10**).