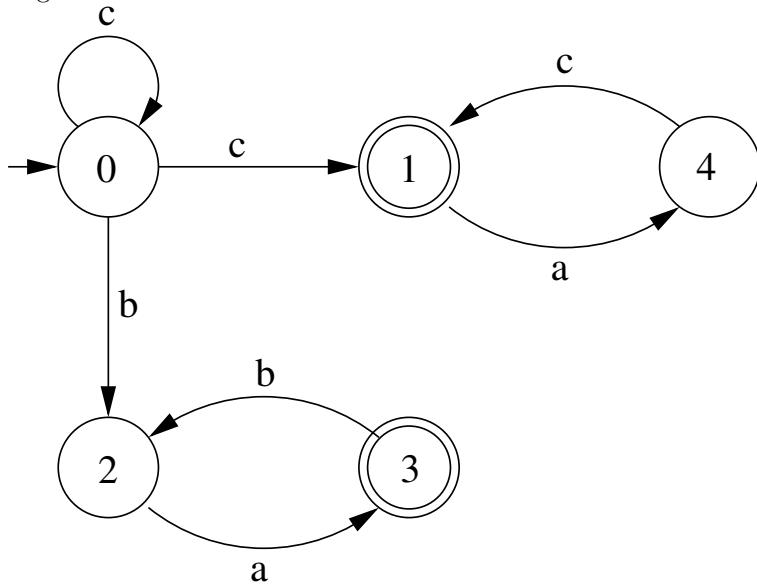


PROGRAMMAZIONE – IV Appello del 17/03/2003

Scrivere **in stampatello** COGNOME, NOME e NUMERO DI MATRICOLA (se conosciuto) su ogni foglio consegnato e sul testo, che va consegnato insieme al compito.

ESERCIZIO 1 (6 punti)

Dato il seguente automa:



- (1) descrivere formalmente il linguaggio riconosciuto dall'automa
- (2) costruire una grammatica *regolare* che genera lo stesso linguaggio.

SOLUZIONE

Il linguaggio riconosciuto dall'automa è il seguente:

$$L = \{c^n(ba)^k \mid n \geq 0, k \geq 1\} \cup \{c^n(ac)^k \mid n \geq 1, k \geq 0\}$$

La grammatica regolare equivalente, ottenuta con l'algoritmo visto a lezione, è la seguente:

```

<0> ::= c <0> | c <1> | b <2> | c
<1> ::= a <4>
<4> ::= c <1> | c
<2> ::= a <3> | a
<3> ::= b <2>
  
```

ESERCIZIO 2 (10 punti)

Data la seguente grammatica:

$\langle S \rangle ::= p \langle S \rangle q \mid \langle T \rangle$
 $\langle T \rangle ::= rr \langle T \rangle q \mid rrq$

- (1) descrivere formalmente il linguaggio generato
- (2) definire un sistema di transizioni $\rightarrow_{r:2}$ che, presa una stringa del linguaggio generato dalla grammatica, restituisca la stessa stringa in cui il numero di caratteri r è stato dimezzato.

Ad esempio: $pprrrrrrqqqq \rightarrow_{r:2} pprrrqqqq$.

Suggerimento: si usi la tecnica Big-Step e si definiscano le regole del sistema di transizioni sulla base delle regole della grammatica (approccio di definizione della semantica guidato dalla sintassi.)

SOLUZIONE

Il linguaggio generato dalla grammatica è il seguente:

$$L = \{p^m r^{2n} q^{m+n} \mid m \geq 0, n \geq 1\}$$

Le regole del sistema di transizioni sono le seguenti

$$\begin{aligned}
 (r1) \quad & \overline{rrq \rightarrow_{r:2} rq} \\
 (r2) \quad & \frac{\alpha \in \{r^{2n} q^n \mid n \geq 1\}, \quad \alpha \rightarrow_{r:2} \beta}{rr \alpha q \rightarrow_{r:2} r \beta q} \\
 (r3) \quad & \frac{\alpha \in L, \quad \alpha \rightarrow_{r:2} \beta}{p \alpha q \rightarrow_{r:2} p \beta q}
 \end{aligned}$$

Si noti che la regola (r2) funziona per due casi sintattici: per un albero di derivazione che ha come radice la categoria sintattica $\langle T \rangle$ e per un albero di derivazione che ha come radice $\langle S \rangle$ e applica subito la regola $\langle S \rangle ::= \langle T \rangle$.

Le configurazioni del sistema sono le stringhe $\Gamma = L \cup \{p^m r^n q^{m+n} \mid m \geq 0, n \geq 1\}$. Le configurazioni finali sono le stringhe $T = \{p^m r^n q^{m+n} \mid m \geq 0, n \geq 1\}$.

ESERCIZIO 3 (4 punti)

Si supponga di estendere la sintassi del Java vista a lezione con la seguente produzione:

`Com ::= Ide = Exp; par Ide = Exp;`

Il comando consente di fare due assegnamenti in parallelo. Ad esempio, l'esecuzione del comando `x = E1; par y = E2;;`, consiste nell'effettuare i due assegnamenti nello stato corrente. Il valore delle espressioni E1 ed E2 deve essere calcolato a partire dallo stato di partenza in modo da evitare interferenze tra i due assegnamenti.

Definire le nuove regole di semantica per trattare il nuovo comando nel modello in cui lo stato è composto solo da una pila di frame σ . Le regole devono essere scritte in modo tale che non si possano applicare se le due variabili x e y che si stanno assegnando in parallelo sono la stessa variabile.

SOLUZIONE

$$(com_{=par}) \quad \frac{x \neq y, \quad \langle E1, \sigma \rangle \rightarrow_{exp} v_1 \quad \langle E2, \sigma \rangle \rightarrow_{exp} v_2}{\langle x = E1; par y = E2;;, \sigma \rangle \rightarrow_{com} \sigma[v_1/x, v_2/y]}$$

ESERCIZIO 4 (5 punti)

Completare la definizione del seguente metodo.

```
public boolean metodo(int[] a)
/** Restituisce true se tutti gli elementi in posizione dispari
 * dell'array a sono numeri pari. Altrimenti restituisce false.
 */
```

SOLUZIONE

```
public boolean metodo(int[] a) {
    // cerco nell'array il primo elemento in posizione dispari che
    // NON e' un numero pari. Se lo trovo restituisco false, altrimenti
    // true.

    boolean trovato = false;
    int i=1; // L'elemento 0 e' in posizione pari e non viene considerato
    while ( i < a.length && !trovato)
        if (a[i] % 2 == 0)
            i = i + 2; // controllo il prossimo elemento in pos dispari
        else
            trovato = true;
    if (trovato)
        return false;
    else
        return true;
}
```

ESERCIZIO 5 (6 punti)

Si consideri il seguente programma

```
prog { class Point {
    public int x;
    public int y;

    public void reset() {
        this.x = 0;
        this.y = 0;
    }
    public void sum(int x, int y) {
        this.x = this.x + x;
        this.y = this.y + y;
    }
} (1)
{
    Point p = new Point;
    p.reset();
    Point q = new Point;
```

```

        q.x = 3; q.y = 5; (2)
        q.sum(q.x, q.y);
        p.sum(q.x, q.y); (3)
    }
}

```

Disegnare l'ambiente delle classi al punto (1) e lo stato (σ e ζ) ai punti (2) e (3) del programma.

SOLUZIONE

Ambiente delle classi al punto (1):

ρ_c

Point	$\{x \rightarrow \bar{\omega}, y \rightarrow \bar{\omega}\}$	ρ_m^{Point}
-------	--	-------------------------

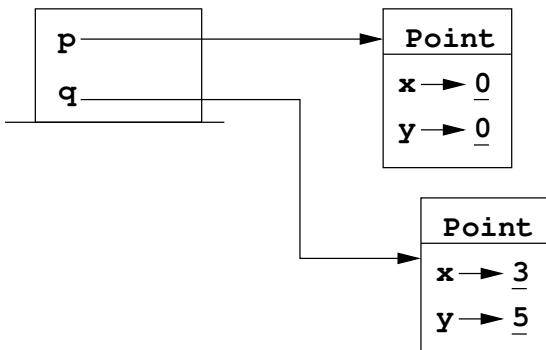
ρ_m^{Point}

reset	()	{this.x = 0; this.y = 0; }
sum	(x, y)	{this.x = this.x + x; this.y = this.y + y; }

Stato al punto (2):

σ

ζ



Stato al punto (3):

