

Cognome & Matricola

NON CONSEGNO

La prova è

- **CLOSED BOOKS:** libri, appunti e quant'altro non sono ammessi;
- **NO ELECTRONICS:** cellulari, palmari e quant'altro non sono ammessi;
- **NO PENCILS & NO MULTICOLOR:** sono valutate solo le risposte scritte a penna e in monocolor.

La prova è composto da 13 esercizi. I primi 12 esercizi richiedono risposte sintetiche, da inserire negli spazi appositamente riservati. Lo svolgimento dell'esercizio 13 va scritto in calce al testo dell'esercizio.

Chi **non vuole** consegnare deve apporre una firma sulla riga “NON CONSEGNO” e restituire il plico.

Gli esercizi sono preceduti da un paragrafo che contiene le definizioni cui si fa riferimento negli esercizi.

Definizioni

- \mathcal{N}_1 : Sia \mathcal{N}_1 lo NFA con stato iniziale A , stato finale E e con la seguente funzione di transizione

	ϵ	a	b
A	$\{B, E\}$	\emptyset	\emptyset
B	$\{C\}$	\emptyset	$\{E\}$
C	\emptyset	$\{D\}$	\emptyset
D	$\{E\}$	\emptyset	$\{B\}$
E	\emptyset	$\{E\}$	$\{A\}$

- \mathcal{D}_1 : Sia \mathcal{D}_1 il DFA con stato iniziale A , stato finale D e con la seguente funzione di transizione

	a	b
A	B	
B	D	C
C	D	
D		B

- \mathcal{G}_1 : Sia \mathcal{G}_1 la seguente grammatica:

$$\begin{aligned}
 S &\rightarrow AaB \mid b \\
 A &\rightarrow BcBaA \mid \epsilon \\
 B &\rightarrow \epsilon
 \end{aligned}$$

- **f1.1:** Sia **f1.1** il file per analizzatore lessicale di cui sono riportate nel seguito le sezioni delle dichiarazioni e delle regole:

```

digit [0-9]
letters [a-z,A-Z]
other [\\*?<>"]
id ({other}|{digit}|{letters})+
%%
({other}":")?(\\)?({id}\\\\)*{id}(\\.{id})? {printf("SI ");}
. {printf("NO ");}
%%

```

f2.1: Sia f2.1 il file per analizzatore lessicale di cui è riportata nel seguito la sezione delle regole:

```

%%
"a"  { return a; }
"b"  { return b; }
"c"  { return c; }
"d"  { return d; }
[0-9]+ { yylval = atoi(yytext); return num; }
[-+\\n] return *yytext;
[ \\t] ;
.      yyerror("invalid character");
%%

```

f2.y: Sia f2.y il file per analizzatore sintattico di cui sono riportate nel seguito le sezioni delle dichiarazioni e delle regole:

```

%token a b c d num
%right a b
%right c d
%%
S:
    S E '\\n' { printf("%d\\n", $2); }
|
;
E:
    num { $$ = $1; }
| E a E { $$ = $1 * $3; }
| E b E { $$ = $1 / $3; }
| E c E { $$ = $1 + $3; }
| E d E { $$ = $1 - $3; }
;
%%

```

f3.y: Sia f3.y il file per analizzatore sintattico di cui sono riportate nel seguito le sezioni delle dichiarazioni e delle regole:

```

%token a b p o c
%%
S : a b p S
  | S A
  | A
;
A : b
  | o S c
;
%%

```

Esercizi

1

Se $\{ww \mid w \in \mathcal{L}((a \mid b)^*)\}$ è un linguaggio regolare rispondere “SI”, altrimenti rispondere “NO”.

2

Se la seguente affermazione è vera rispondere “VERO”, altrimenti rispondere “FALSO”: “*Se i linguaggi \mathcal{L}_1 e \mathcal{L}_2 sono entrambi regolari allora $\mathcal{L}_1 \cup \mathcal{L}_2$ è regolare.*”

3

Sia $r = b^* \mid b^*a(\epsilon \mid a \mid b)^*$ e sia \mathcal{D} il DFA minimo per il riconoscimento di $\mathcal{L}(r)$. Dire quanti stati ha \mathcal{D} e quanti di questi stati sono finali .

4

Chiamiamo \mathcal{D} il DFA ottenuto da \mathcal{N}_1 per subset construction, Q lo stato iniziale di \mathcal{D} , Q_{ab} lo stato di \mathcal{D} che si raggiunge da Q tramite il cammino ab . Dire a quale sottoinsieme degli stati di \mathcal{N}_1 corrisponde Q_{ab} .

5

Chiamiamo \mathcal{D}_m il DFA ottenuto per minimizzazione di \mathcal{D}_1 , P lo stato iniziale di \mathcal{D}_m , P_{abab} lo stato di \mathcal{D}_m che si raggiunge da P tramite il cammino $abab$. Dire a quale sottoinsieme degli stati di \mathcal{D}_1 corrisponde P_{abab} .

6

Scrivere l'intera riga della tabella di parsing LL(1) per \mathcal{G}_1 relativa al non-terminale B .

7

Chiamiamo \mathcal{A} l'automa LR(0) per il parsing SLR(1) di \mathcal{G}_1 , I lo stato iniziale di \mathcal{A} , I_{BcBa} lo stato di \mathcal{A} che si raggiunge da I tramite il cammino $BcBa$. Se la tabella di parsing SLR(1) per \mathcal{G}_1 non ha alcun conflitto nello stato I_{BcBa} , rispondere "NO". Altrimenti specificare di che conflitto si tratta indicando quali entry dello stato contengono conflitti e quali reduce sono coinvolte.

8

Chiamiamo \mathcal{A} l'automa LR(1) per il parsing LR(1) di \mathcal{G}_1 , J lo stato iniziale di \mathcal{A} , J_{Aa} lo stato di \mathcal{A} che si raggiunge da J tramite il cammino Aa . Elencare gli item LR(1) che appartengono a J_{Aa} .

9

Chiamiamo \mathcal{A} l'automa LR(0) per \mathcal{G}_1 , H lo stato iniziale di \mathcal{A} , H_{BcBaBc} lo stato di \mathcal{A} che si raggiunge da H tramite il cammino $BcBaBc$, T la tabella di parsing LALR(1) per \mathcal{G}_1 . Se non ci sono riduzioni nello stato H_{BcBaBc} di T , rispondere "NO". Altrimenti dire quali entry dello stato H_{BcBaBc} di T contengono riduzioni e specificare di che riduzioni si tratta.

10

Sia `exec` l'eseguibile ottenuto utilizzando l'analizzatore lessicale su `f1.1`. Dire quale output viene prodotto applicando `exec` al seguente input:

DC:\home27>

11

Sia `exec` l'eseguibile ottenuto utilizzando l'analizzatore sintattico su `f2.y` e l'analizzatore lessicale su `f2.1`. Dire quale output viene prodotto applicando `exec` al seguente input:

1c2c3a4

12

Si consideri il parser ottenuto dall'analizzatore sintattico per `f3.y`. Tra le informazioni di dettaglio fornite dal parser troviamo, tra l'altro, le seguenti:

State 12 conflicts: 2 shift/reduce

State 12

1 S: a b p S .

2 | S . A

4 A: . b

5 | . o S c

Fornire una stringa appartenente al linguaggio generato dalla grammatica il cui parsing porterebbe ad uno dei conflitti menzionati.

13

Sia \mathcal{G} la seguente grammatica:

$$\begin{aligned} S &\rightarrow Aa \mid Bb \\ A &\rightarrow aAb \mid ab \\ B &\rightarrow aBbb \mid abb \end{aligned}$$

Evitando di ricorrere alla computazione della tabella di parsing, spiegare perché \mathcal{G} certamente non è LR(1).