

1. Konstruisati MDKA za izraz $(a*bb)+ab$ i zatim napisati C program koji proverava da li uneta niska pripada ovom jeziku.
2. *LL(1)* gramatikom opisati uređene liste u HTML-u, a zatim napraviti perl skript koji metodom rekursivnog spusta proverava da li je na ulazu zadata ispravna lista.

```
<ol>
  <li>Prvi element </li>
  <li>Drugi element </li>
  <li>Treci element </li>
</ol>
```

3. Napisati interpetator za jezik u kome je osnovni tip tip *Matrix*. Jezik je strogo tipiziran, tako da zahteva deklaraciju svih promenljivih, pre njihove upotrebe.

- (a) *Matrix* predstavlja tip podatka kojim se predstavljaju matrice proizvoljne dimenzije. Imena promenljivih ovog tipa počinju malim slovom za kojim eventualno mogu da slede cifre. Omogućiti deklarisanje ovog tipa na sledeći način: (deklaracijom mogu i da se inicijalizuju matrice)

Matrix m = [1,2,3;4,5,6;7,8,9], m1 = [1;2;3], m2 = [-1,-2,-3;-4,-5,-6;-7,-8,-9], m3;

- (b) Realizovati štampanje matrice i promenu vrednosti već deklarisanim promenljivama. Dozvoljene operacije nad matricama su sabiranje, oduzimanje, množenje i množenje konstantom. Omogućiti ispis nekog izraza sa matricama, kao i postavljanje promenljive na vrednost nekog izraza.

```
print m;
| 1 2 3 |
| 4 5 6 |
| 7 8 9 |

print m + m2;
| 0 0 0 |
| 0 0 0 |
| 0 0 0 |

m1 = [1,2,3];
print m1 * m;
| 30 36 42 |
Matrix w = 2*[-0.5;-0.5];
print w;
| -1 |
| -1 |
```

- (c) *transformation* predstavlja tip podatka kojim se predstavljaju transformacije matrica. Tipovi transformacija su *shifth* i *shiftv*, koji od matrice na koju se primenjuju daju, redom, matricu koja je dobijena cikličnim pomeranjem njenih elemenata, po vrstama (ulevo ili udesno), i pomeranjem elemenata po kolonama (na gore, ili na dole). Omogućiti deklaraciju, inicijalizaciju, promenu vrednosti i primenu transformacije na matrice. Imena promenljivih koje predstavljaju transformacije počinju znakom \$, pa zatim malim slovom, koje eventualno prate neke cifre.

```
transformation $f = shifth(1), $g;
w = $f(m);
print w;
| 3 1 2 |
| 6 4 5 |
| 9 7 8 |

$g = shiftv(2);
m1 = $g(m);
print m1;
| 4 5 6 |
| 7 8 9 |
| 1 2 3 |

$g = shiftv(-1);
print $g(m);
| 4 5 6 |
| 7 8 9 |
| 1 2 3 |
```

- (d) Omogućiti kompoziciju transformacija na sledeći način:

```
transformation $y = shifth(1) * $g * shifth(1);
print $y(m);
| 5 6 4 |
| 8 9 7 |
| 2 3 1 |
```