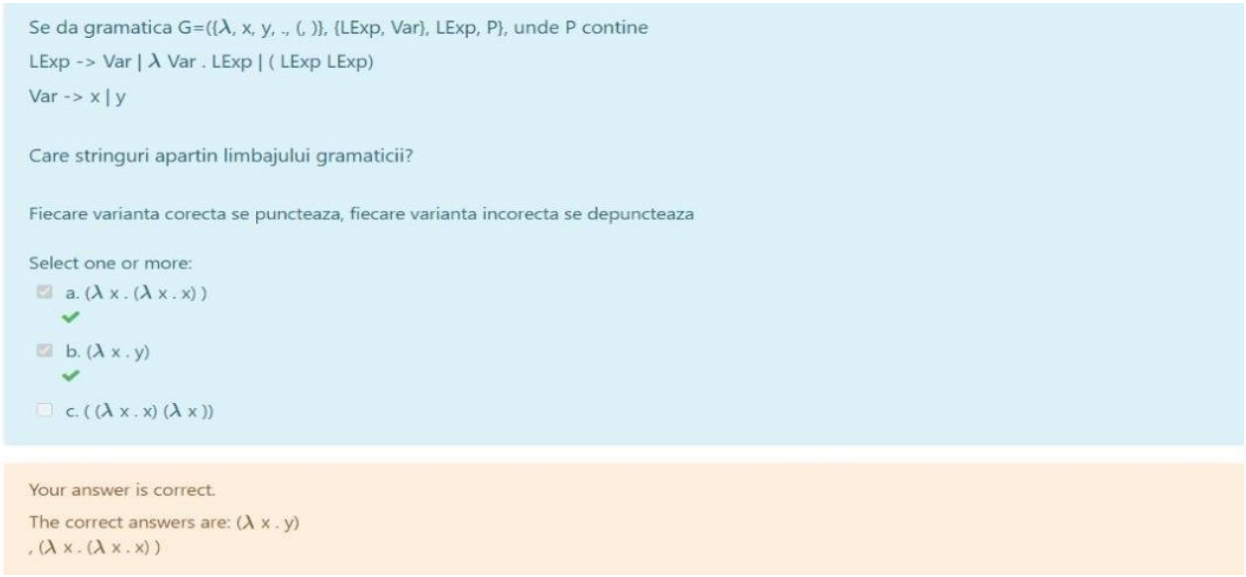
Se da gramatica G=({λ, x, y, ., (,)}, {LExp, Var}, LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp)

Var -> x | y

Care stringuri apartin limbajului gramaticii ?

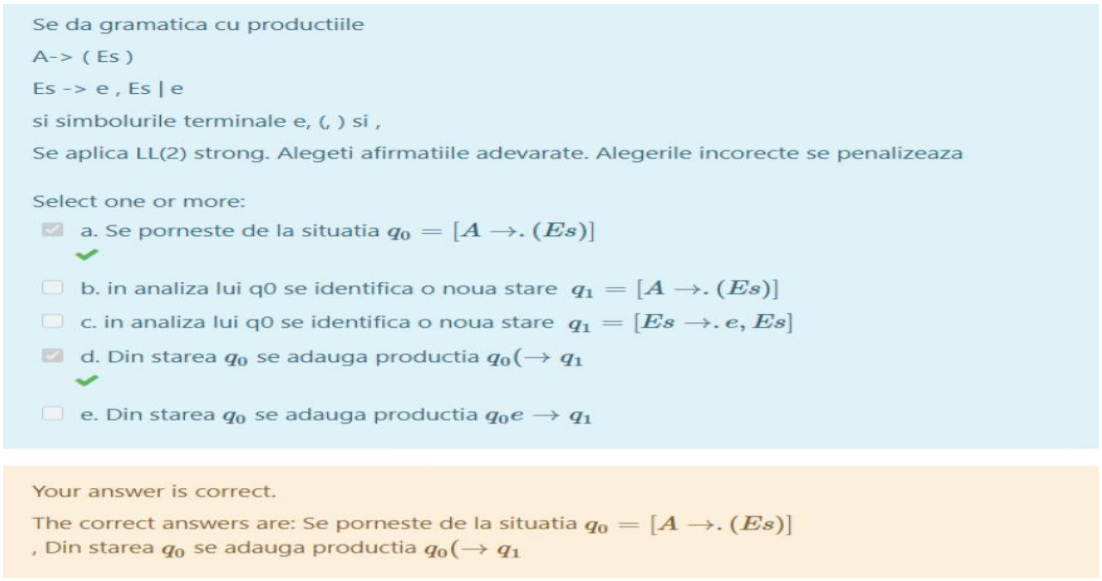


Se da gramatica cu productiile

A -> (Es) Es -> e , Es | e

Si simbolurile terminale e, (,) si ,

Se aplica LL(s) string . Alegeti afirmatiile adevarate .



Care dintre expresiile regulate exprima limbajul {w| w contine pe orice pozitie para 0 }

Care dintre expresiile regulate exprima limbajul {w| w contine pe orice pozitie para 0}

Alegerile gresite se penalizeaza

Select one or more:

- [ ] a. (0+1)*0(1+0)*
- [ ] b. (101)*
- [x] c. ((0+1)0) * (0+1+ε) ✔
- [x] d. ((0+1)0) *1 ✖

Your answer is partially correct.

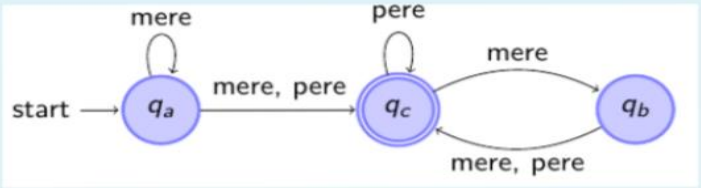
You have selected too many options.

The correct answer is: ((0+1)0) * (0+1+ε)

Ce fel de automat este?

Ce fel de automat este?

$$D_1 = (\{mere, pere\}, \{q_a, q_c, q_b\}, \delta, q_a, \{q_c\})$$



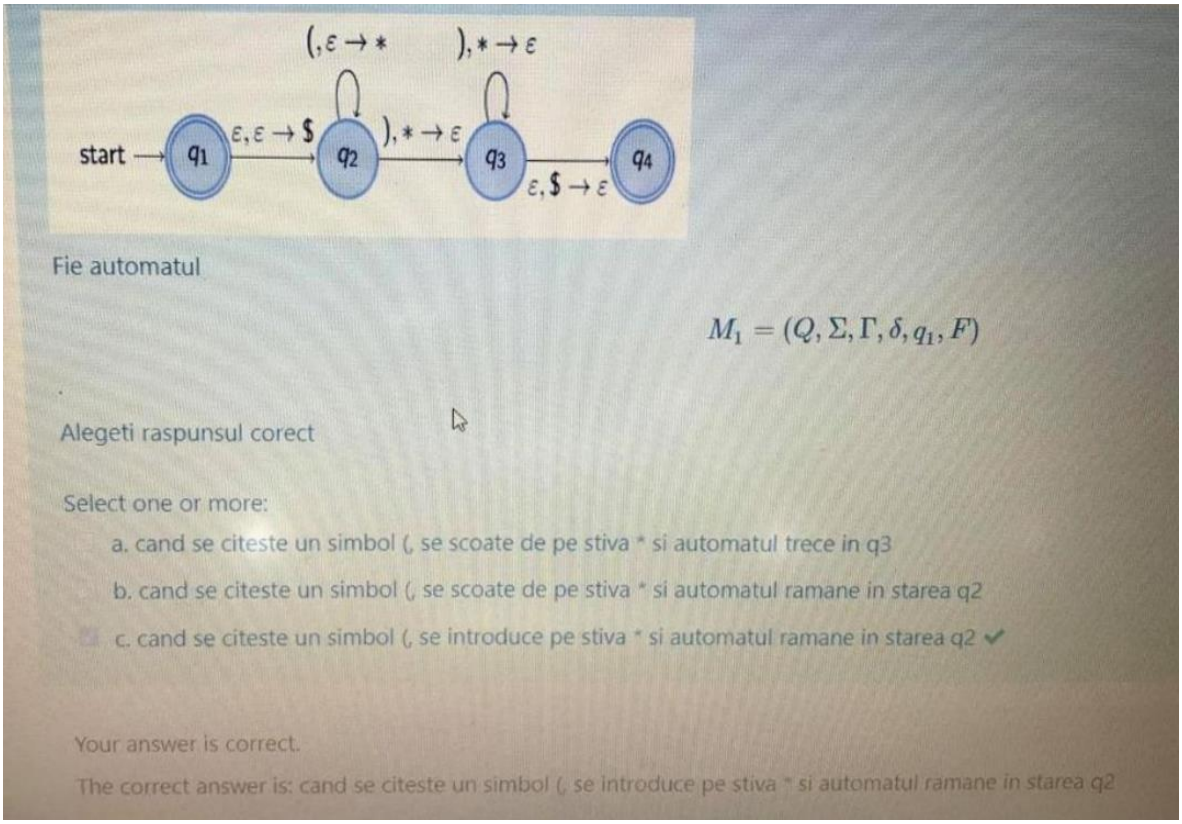
Select one or more:

- [x] a. automat finit nedeterminist ✔
- [ ] b. automat finit determinist
- [ ] c. automat stiva

Your answer is correct.

The correct answer is: automat finit nedeterminist

Fie automatul



Ce fel de automat este ?

Ce fel de automat este?

$$D_1 = (\{mere, pere\}, \{q_a, q_c, q_b\}, \delta, q_a, \{q_c\})$$

	δ	mere	pere
→	q_a	$\{q_a, q_b\}$	q_c
*	q_c	q_b	q_c
	q_b	q_c	q_c

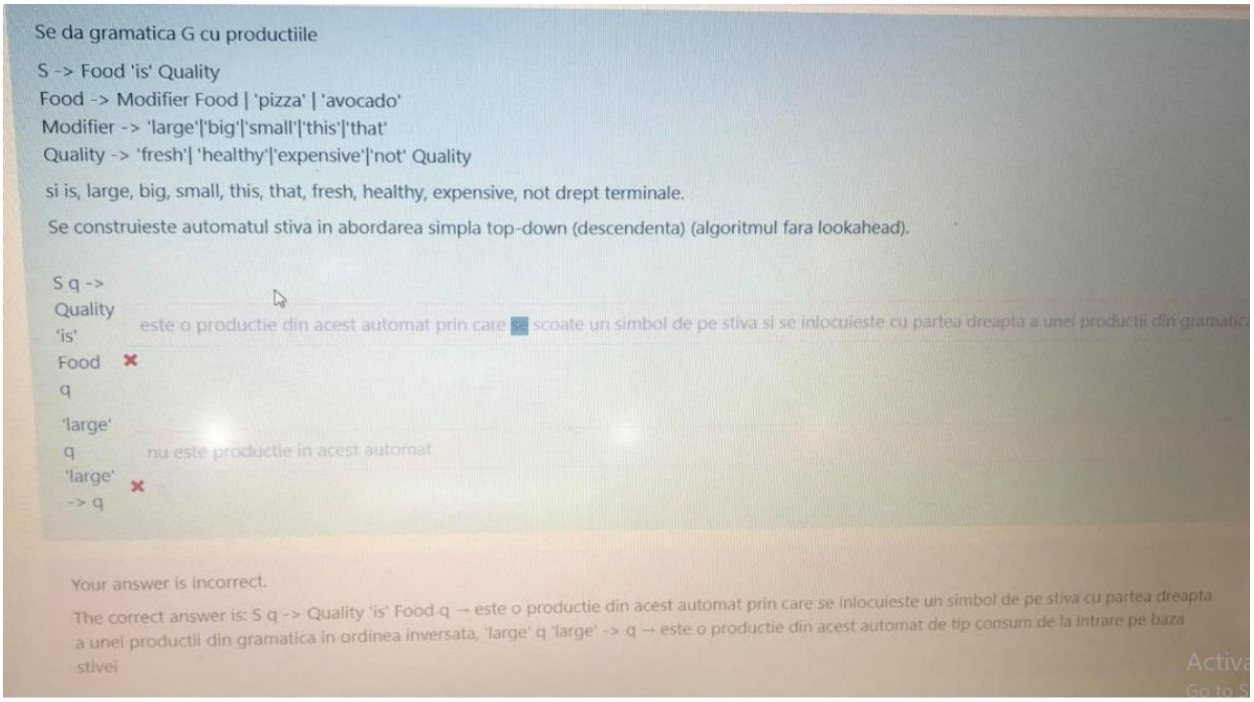
Select one or more:

- [ ] a. automat finint determinist
- [ ] b. automat stiva
- [x] c. automat finit nedeterminist ✔

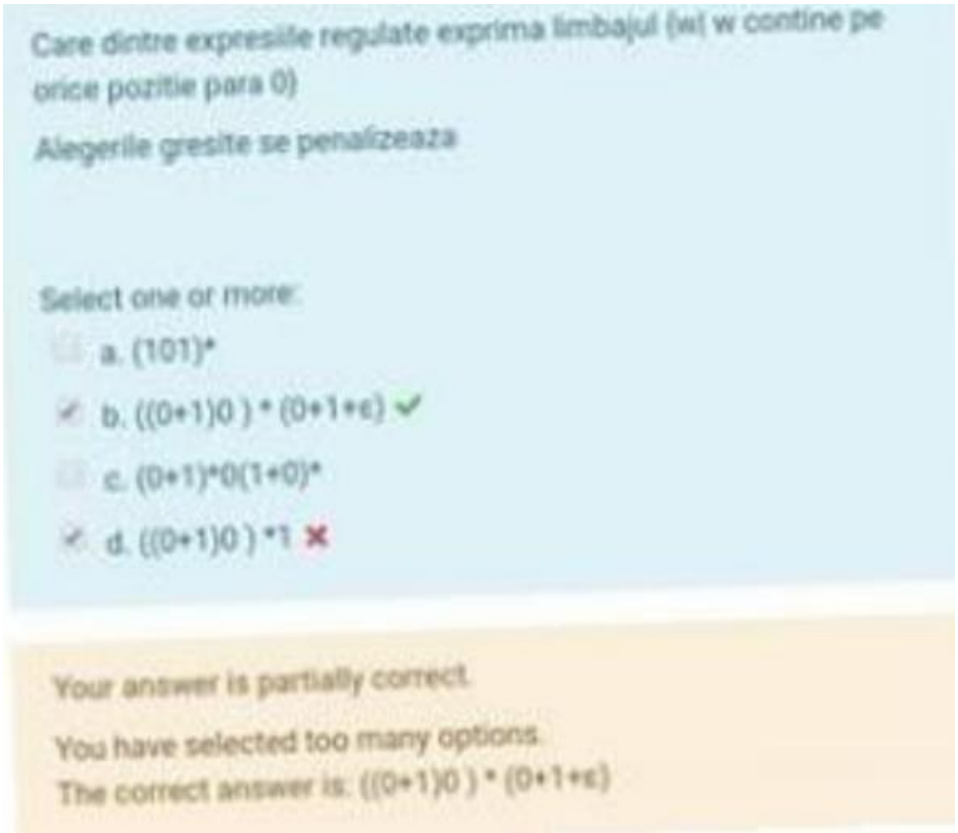
Your answer is correct.

The correct answer is: automat finit nedeterminist

Se da gramatica cu productiile , food modifier quality stiva top-down fara lookahead



Care dintre expresiile regulate exprima limbajul {w| w contine pe orice pozitie para 0 }



Se dau gramaticile G1=({e, ‘,’}, {P,Ps}, P2, Productii1}, G2=({e, ‘,’}, {P,P2},Ps, Productii2},

Se dau gramaticile G1=({e, ','}, {P, Ps}, Ps, Productii1}, G2=({e, ','}, {P, Ps}, Ps, Productii2},

Productii1 = { Ps-> P Ps | ε

P-> e ','}

Productii2 = { Ps-> P Ps | P

P-> e ','}

Care afirmatii sunt adevarate?

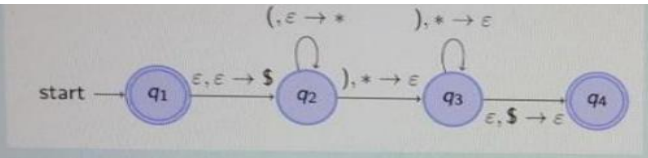
e,e,e apartine — nici L(G1) nici L(G2) ✔

e,e,e, apartine — atat L(G1), cat si L(G2) ✔

Your answer is correct.

The correct answer is: e,e,e apartine → nici L(G1) nici L(G2), e,e,e, apartine → atat L(G1), cat si L(G2)

Fie automatul



Fie automatul

$$M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$$

Alegeti raspunsul corect

Select one or more:

- a. cand se citeste un simbol (se scoate de pe stiva * si automatul trece in q3
- b. cand se citeste un simbol (se scoate de pe stiva * si automatul ramane in starea q2
- c. cand se citeste un simbol (se introduce pe stiva * si automatul ramane in starea q2 ✔

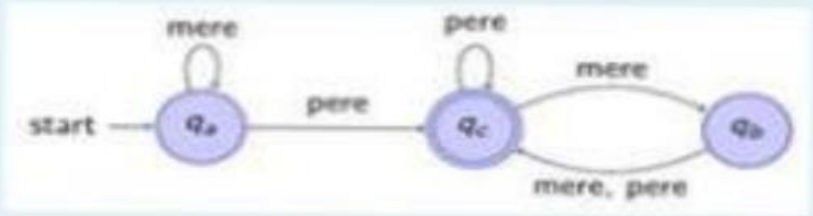
Your answer is correct.

The correct answer is: cand se citeste un simbol (se introduce pe stiva * si automatul ramane in starea q2

Ce fel de automat este ?

Ce fel de automat este?

$$D_1 = (\{mere, pere\}, \{q_a, q_c, q_b\}, \delta, q_a, \{q_c\})$$



Select one or more:

- a. automat finit determinist ✔
- b. automat finit nedeterminist
- c. automat stiva

Your answer is correct.

The correct answer is: automat finit determinist

Se da gramatica G cu productiile

Se da gramatica G cu productiile

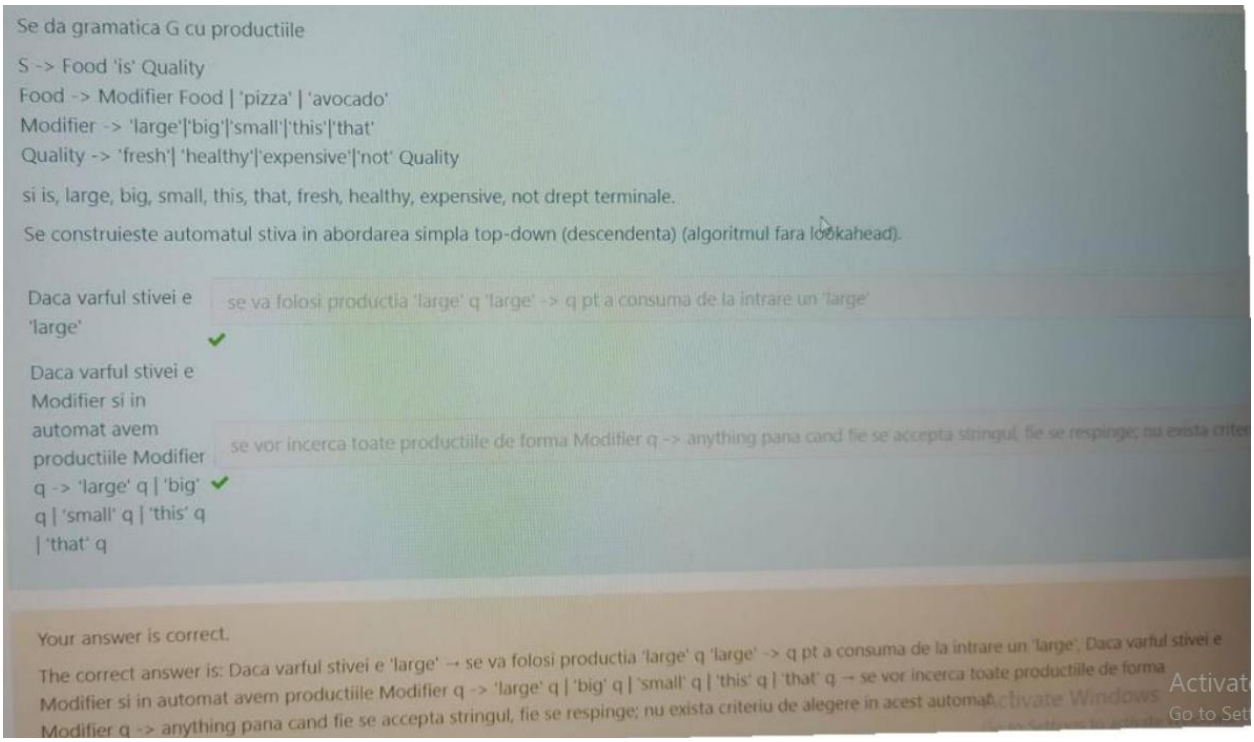
S-> Food ‘is’ Quality

Food -> Modifier Food | ‘pizza’ | ‘avocado’

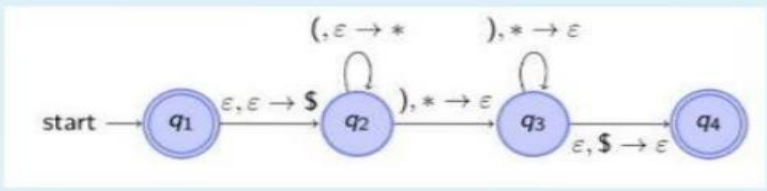
Modifier -> ‘large’ | ‘big’ | ‘small’ | ‘this’ | ’that’

Quality -> ‘fresh’ | ‘healthy’ | ‘expensive’ | ‘not’ Quality

Si is , large, big, small, this, that , fresh , healthy, expensive , not drept terminale .



Fie automatul



Fie automatul

$$M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$$

.

Alegeti raspunsul corect

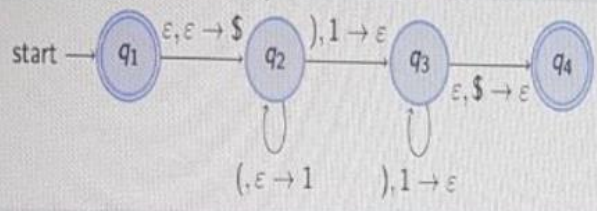
Select one or more:

- a. este un automat finit determinist ✗
- b. este un automat finit nedeterminist
- c. este un automat stiva ✓

Your answer is incorrect.

The correct answer is: este un automat stiva

Fie automatul



Fie automatul

$$M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$$

.

Alegeti raspunsul corect

Select one or more:

- a. in starea q2 citeste) doar daca poate scoate de pe stiva (
- b. in starea q2 citeste) doar daca poate scoate de pe stiva un 1, altfel stringul e respins ✓
- c. in starea q2 citeste) doar daca poate scoate de pe stiva un 1, altfel revine in q1

Your answer is correct.

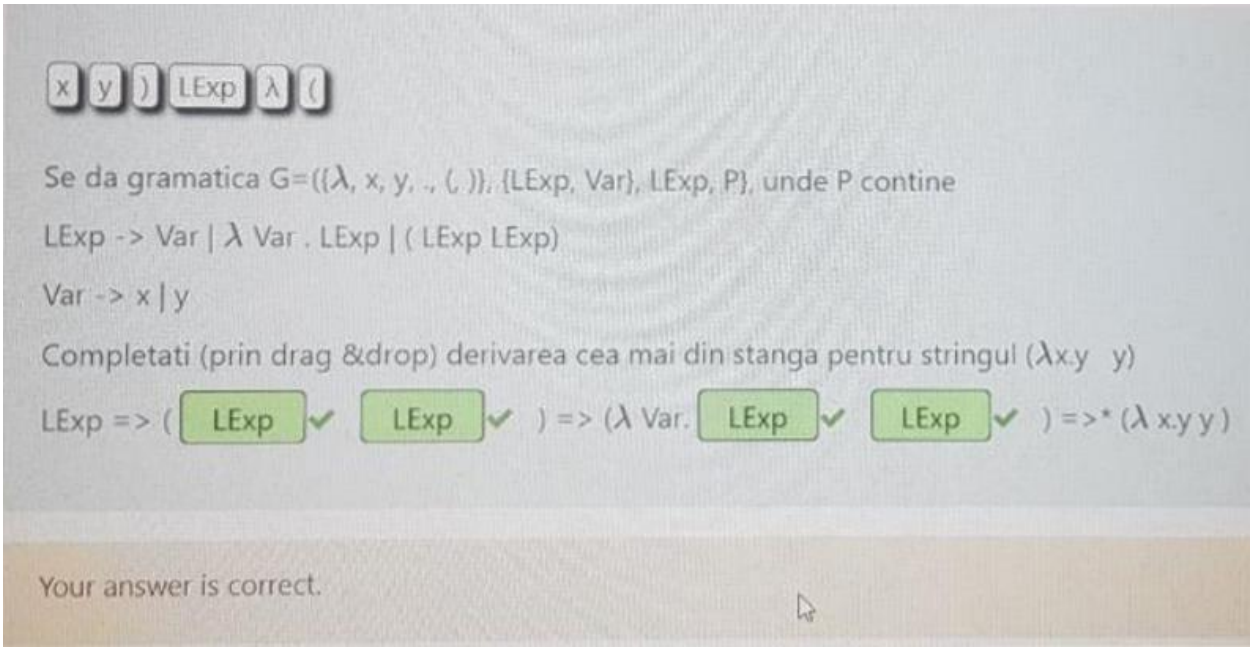
The correct answer is: in starea q2 citeste) doar daca poate scoate de pe stiva un 1, altfel stringul e respins

Se da gramatica G=({ λ, x, y, ., (,)}, {LExp, Var}, {LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp) .

Var -> x | y

Completati (prin drag & drop) derivarea cea mai din stanga pentru stringul (λx.y y)

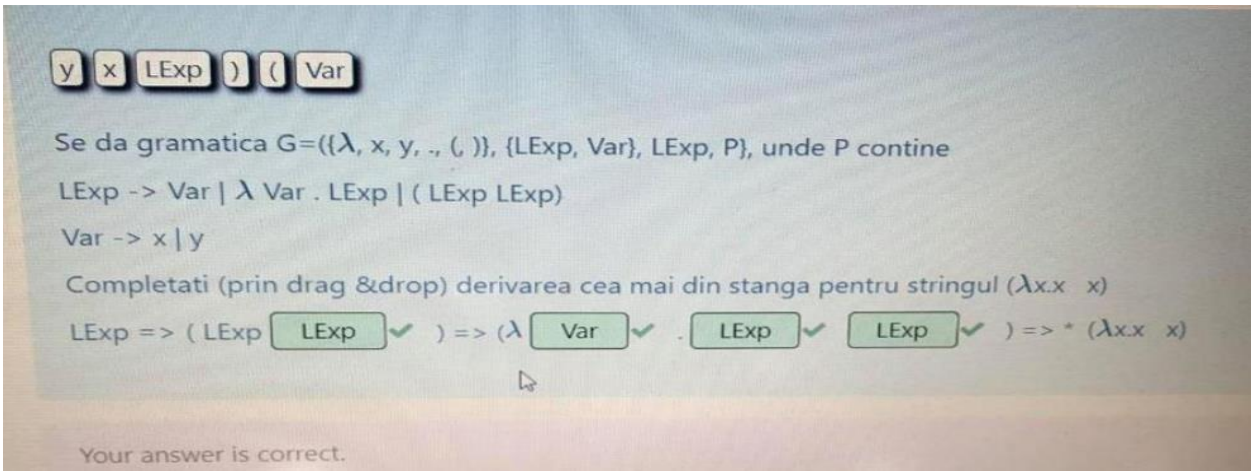


Se da gramatica G=({ λ, x, y, ., (,)}, {LExp, Var}, {LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp) .

Var -> x | y

Completati (prin drag & drop) derivarea cea mai din stanga pentru stringul (λx.x x)



Se da gramatica G cu productiile

forst -> FOR vardecl ‘=’ E ‘to’ stepE sts

stepE -> STEP E| epsilon

sts -> ‘S’ sts| epsilon

STEP -> ‘step’

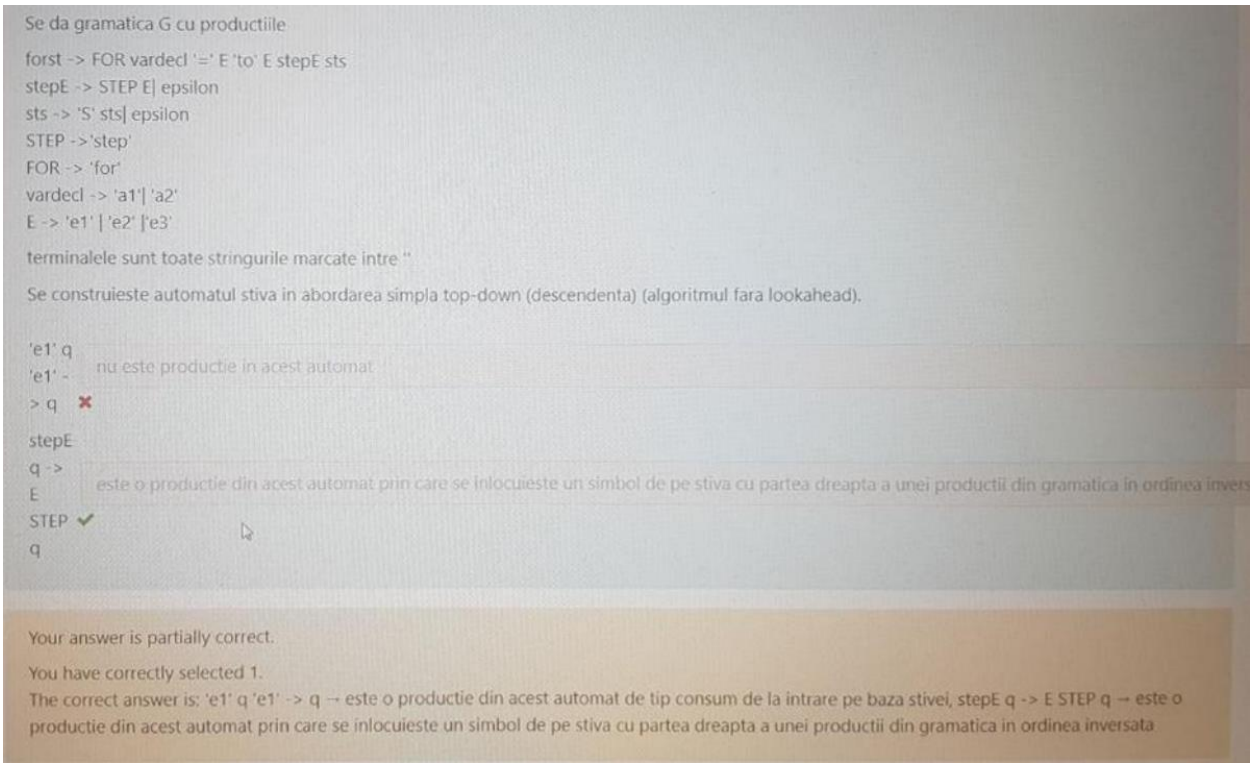
FOR -> ’for’

Vardecl -> ‘a1’|’a2’

E -> ‘e1’ | ‘e2’ | ‘e3’

Terminalele sunt toate stringurile marcate intre ‘’

Se construieste automatul stiva in abordarea simple top-down



Se da gramatica cu productiile

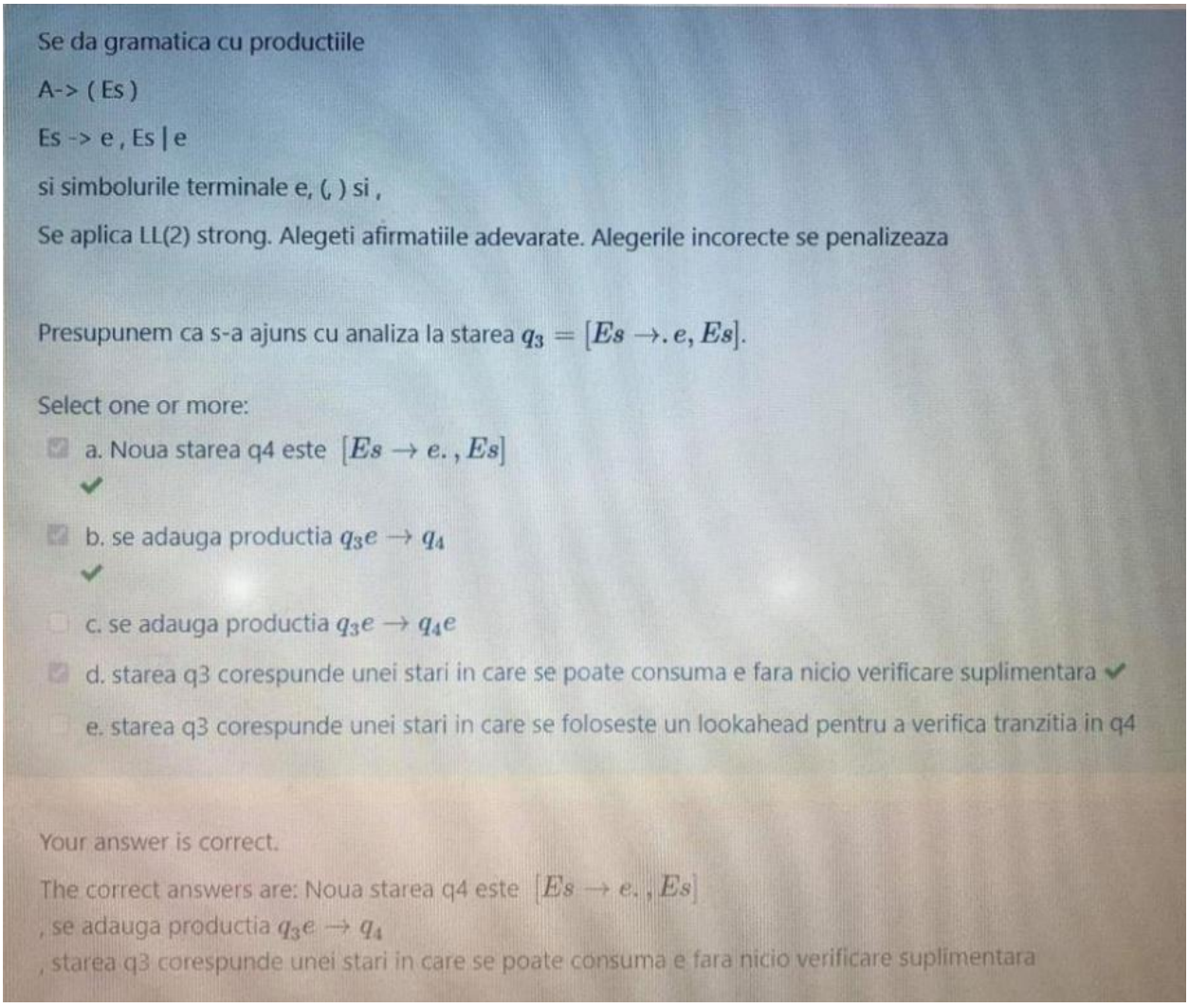
A-> (Es)

Es -> e , Es | e

Si simbolurile terminale e, (,) si ,

Se aplica LL(2) strong. Alegeti afirmatiile adevarate.

Presupunem ca s-a ajuns cu analiza la starea q3 = [Es -> .e, Es].



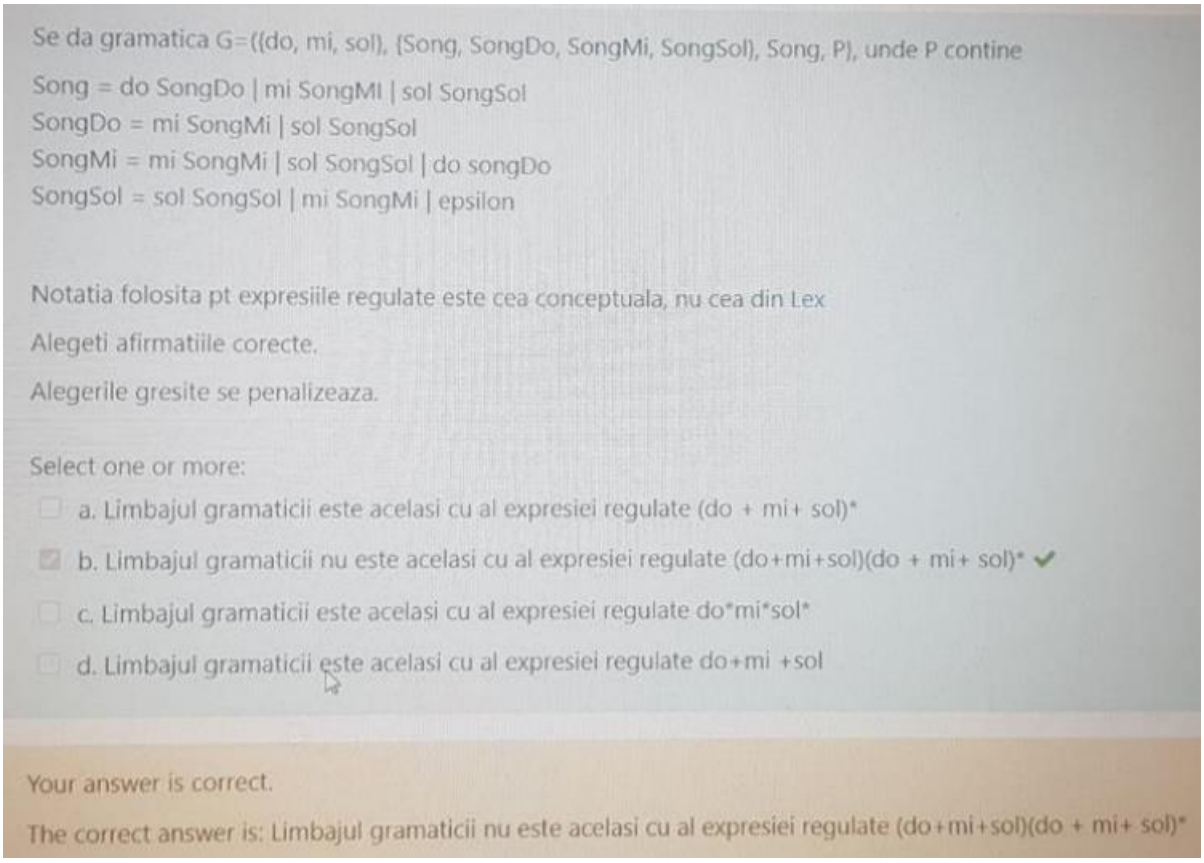
Se da gramatica G=({do, mi, sol}, {Song, SongDo, SongMi, SongSol}, Song, P}, unde P contine

Song = do SongDo | mi SongMi | sol SongSol

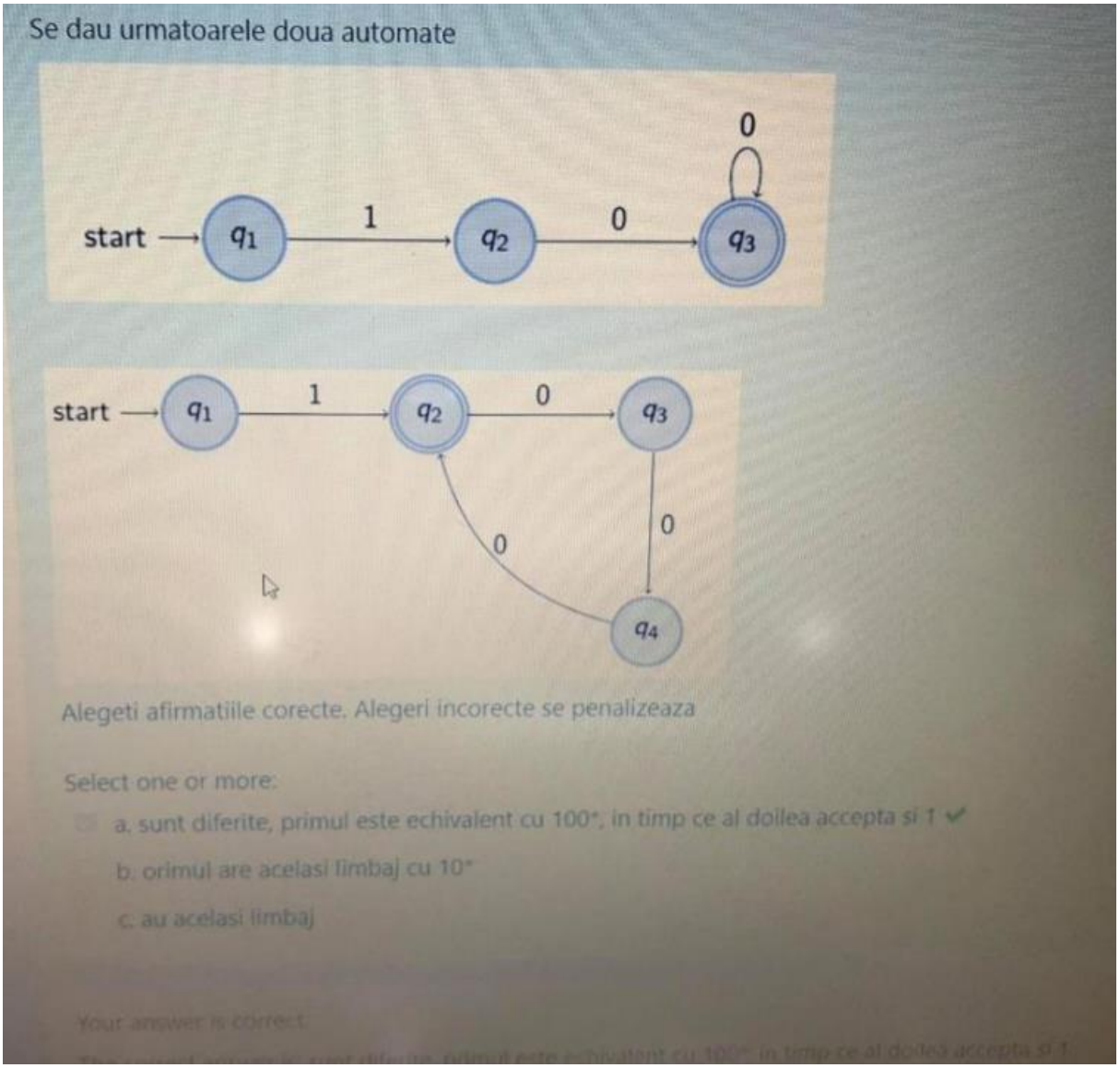
SongDo = mi SongMi | sol SongSol

SongMi = mi SongMi | sol SongSol | do SongDo

SongSol = sol SongSol | mi SongMi | epsilon



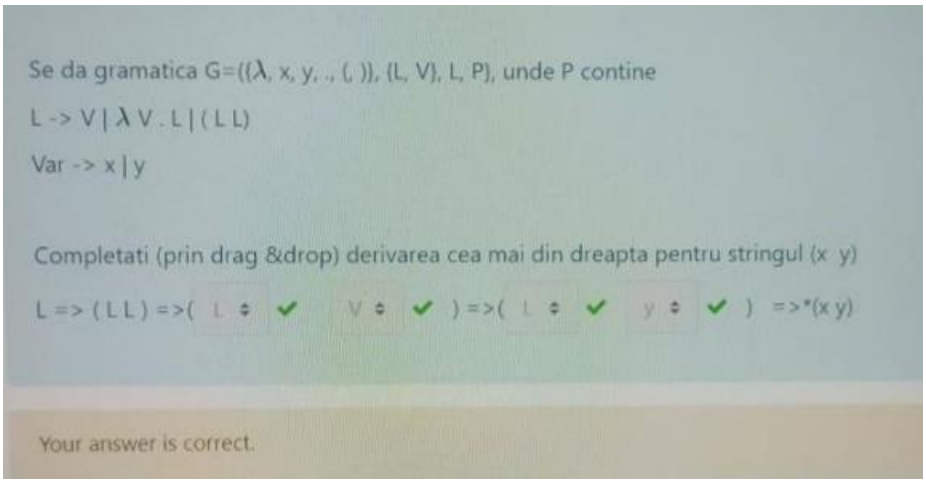
Se dau urmatoarele doua automate



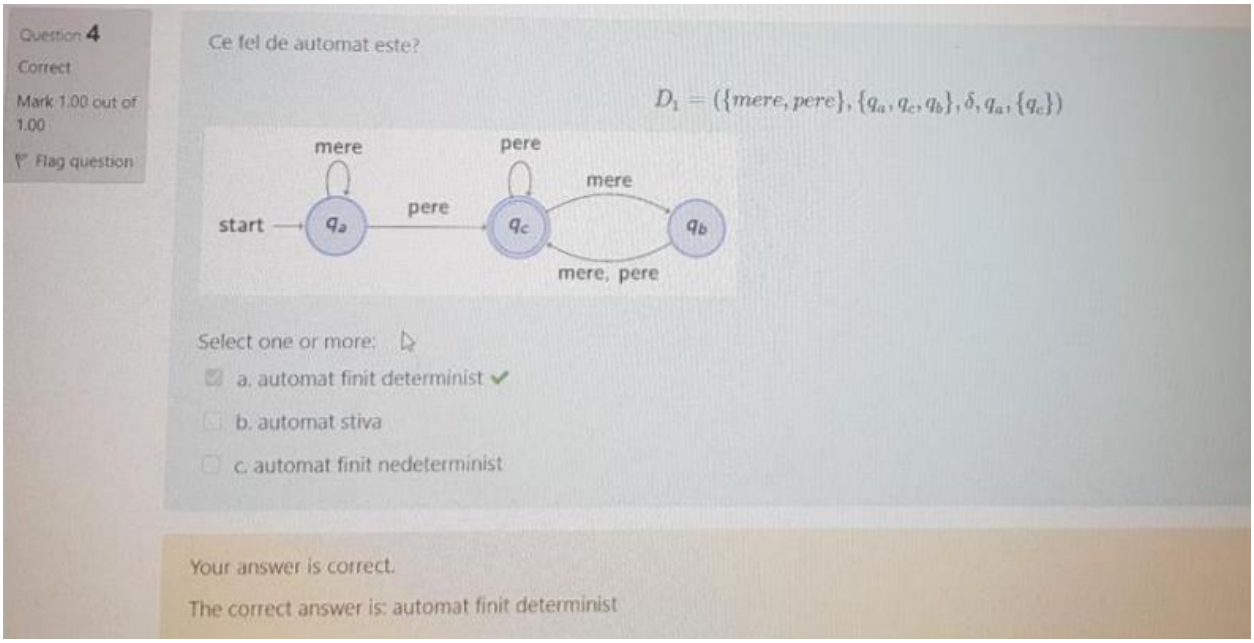
Se da gramatica G=({ λ, x, y, ., (,)}, {L, V}, L, P}, unde P contine

L -> V | λ V . L | (L L)

Var -> x | y



Ce fel de automat este?



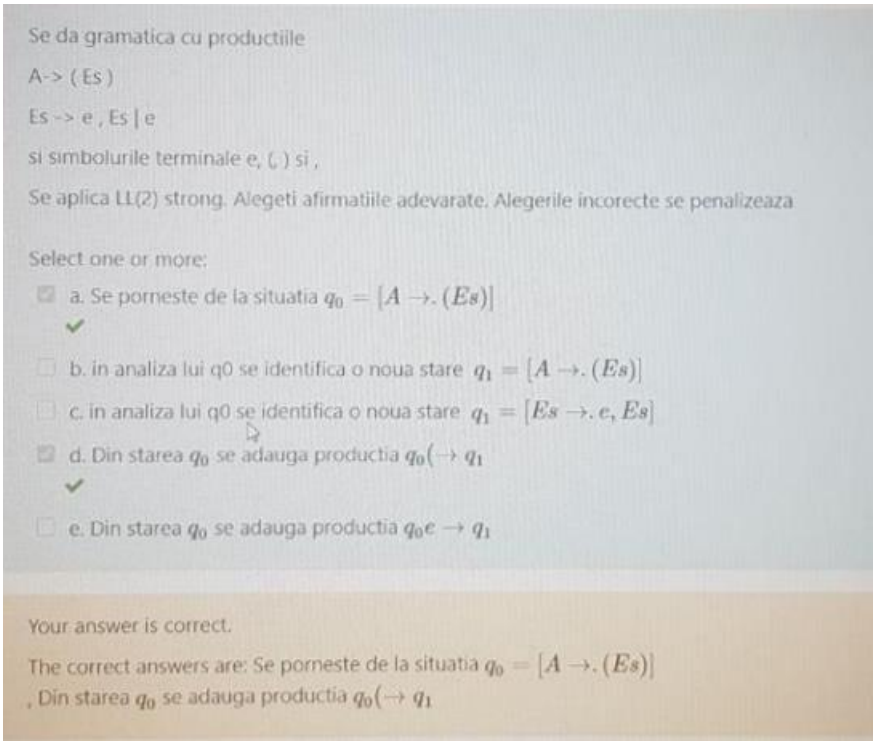
Se da gramatica cu productiile

A -> (Es)

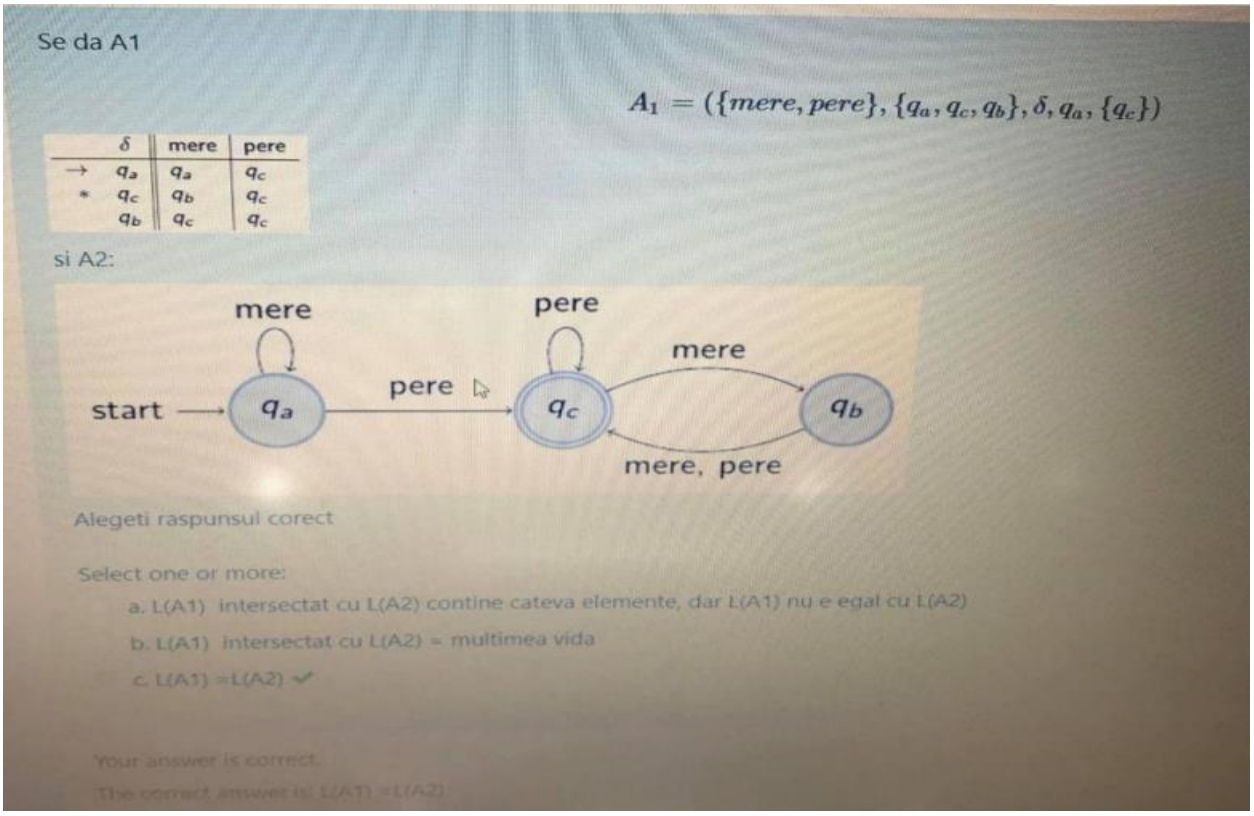
Es -> e , Es | e

Si simbolurile iterminale e, (,) si ,

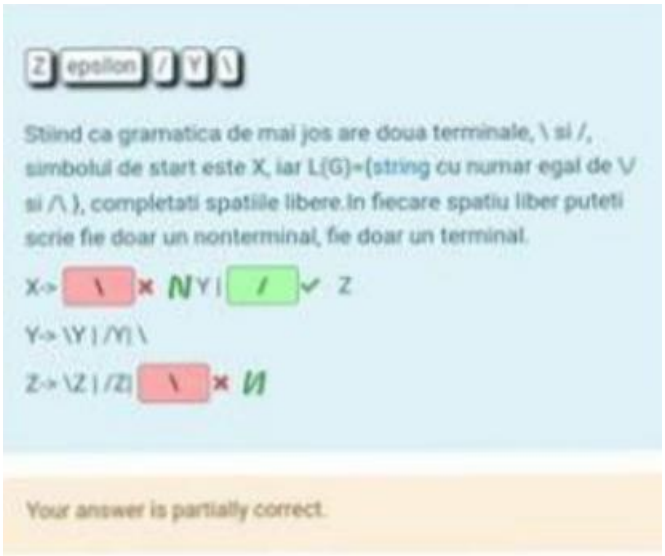
Se aplica LL(2) strong.



Se da A1



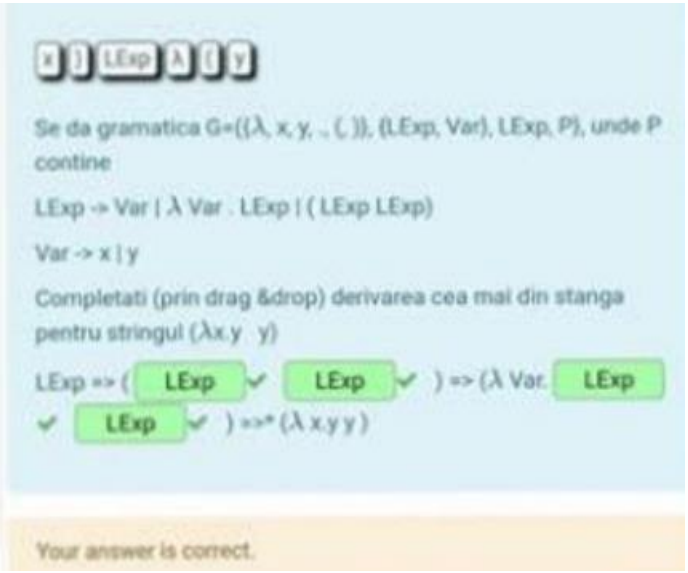
Stiind ca gramatica de mai jos are doua terminale , \ si / , simbolul de start este X , iar L(g) = (string cu numar egal de V si ^}, completati spatiile libere , in fiecare spatiu liber puteti scrie fie doar un nonterminal, fie doar un terminal .



Se da gramatica G=({ λ, x, y, ., (,)}, {LExp, Var} ,LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp)

Var -> x|y



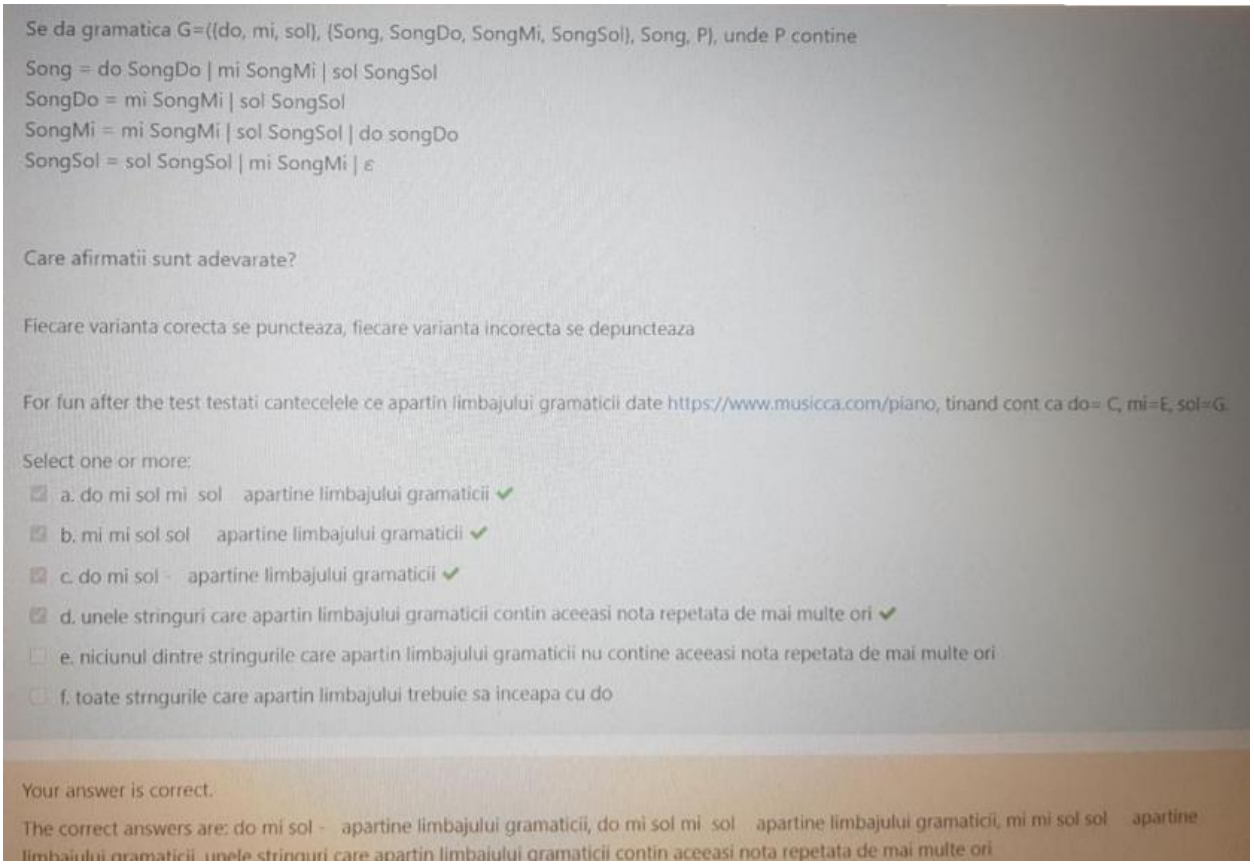
Se da gramatica G=({do, mi, sol}, {Song, SongDo, SongMi, SongSol}, Song, P}, unde P contine

Song = do SongDo | mi SongMi | sol SongSol

SongDo = mi SongMi | sol SongSol

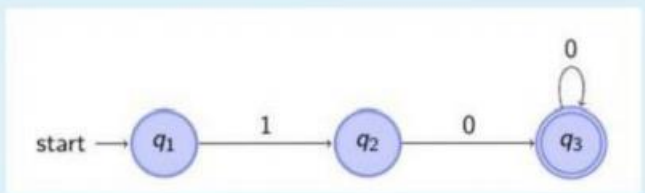
SongMi = mi SongMi | sol SongSol | do SongDo

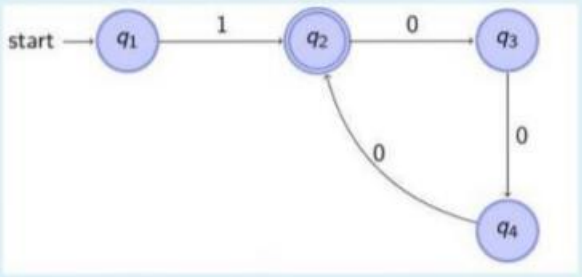
SongSol = sol SongSol | mi SongMi | epsilon



Se dau urmatoarele doua automate

Se dau urmatoarele doua automate





Alegeti afirmatiile corecte. Alegeri incorecte se penalizeaza

Select one or more:

- a. orimul are acelasi limbaj cu 10*
- b. au acelasi limbaj
- c. sunt diferite, primul este echivalent cu 100*, in timp ce al doilea accepta si 1 ✔

Se da gramatica G=({ λ, x, y, ., (,)}, {L, V}, L, P}, unde P contine

L -> V | λ V . L | (L L)

Var -> x | y

Se da gramatica G=({λ, x, y, ., (,)}, {L, V}, L, P}, unde P contine

L -> V | λ V . L | (L L)

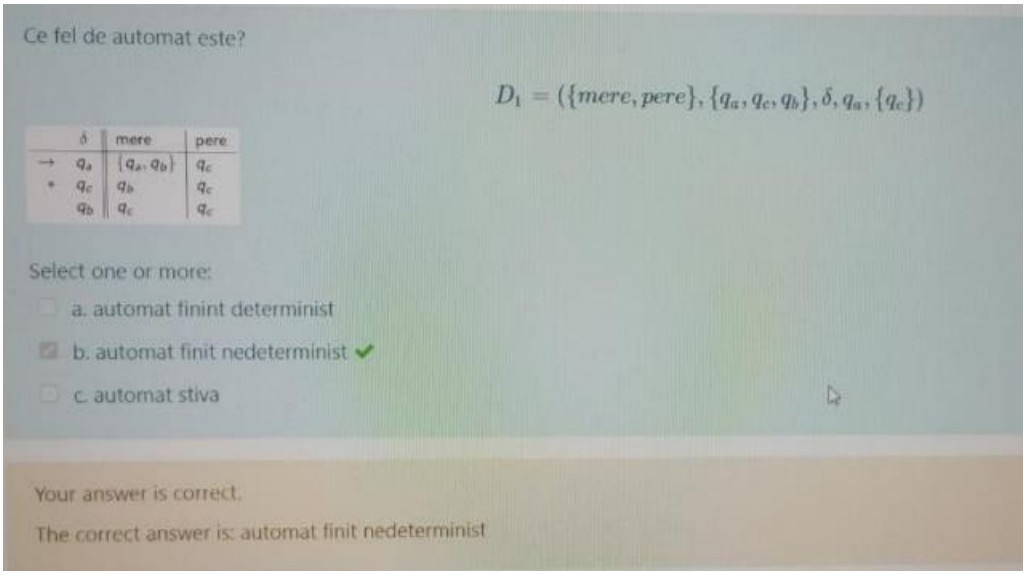
Var -> x | y

Completati (prin drag &drop) derivarea cea mai din dreapta pentru stringul (x y)

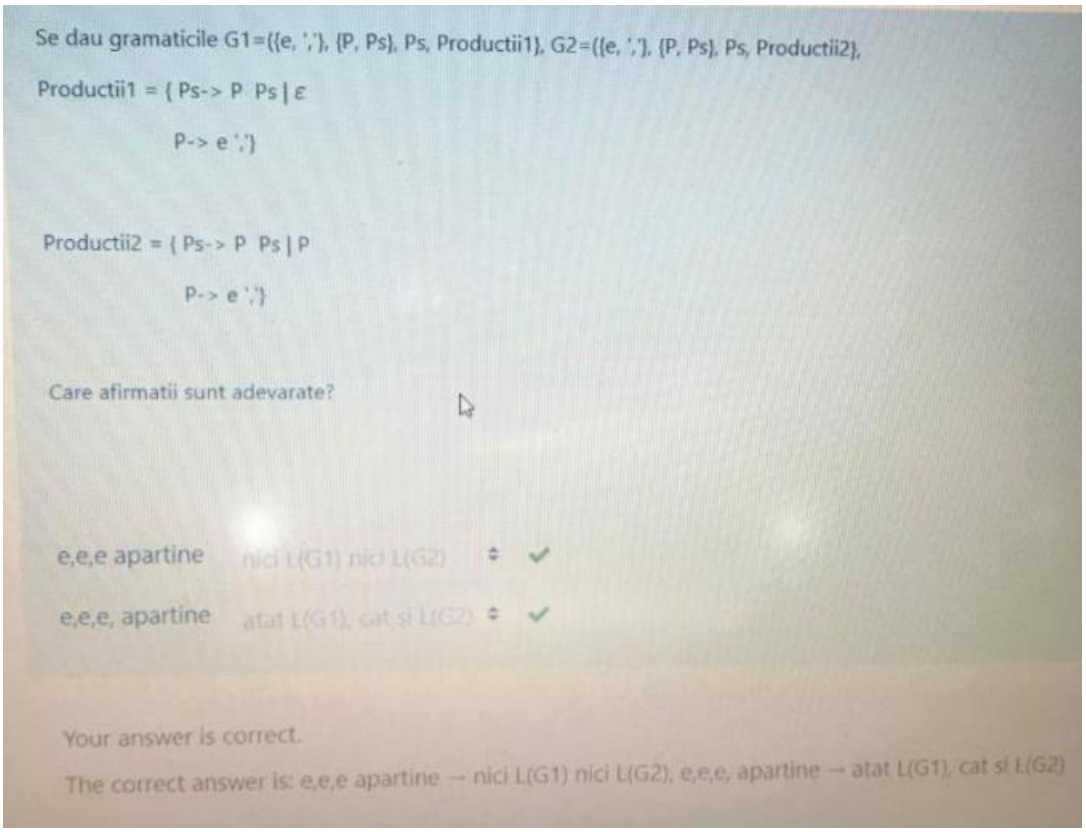
L => (L L) =>(L ✔ V ✔) =>(L ✔ y ✔) =>*(x y)

Your answer is correct.

Ce fel de automat este?



Se dau gramaticile G1 = ({e, ‘,’}, {P, P2}, Ps, Productii1}, G2={{e, ‘,’}, {P, Ps}, Ps, Productii2},



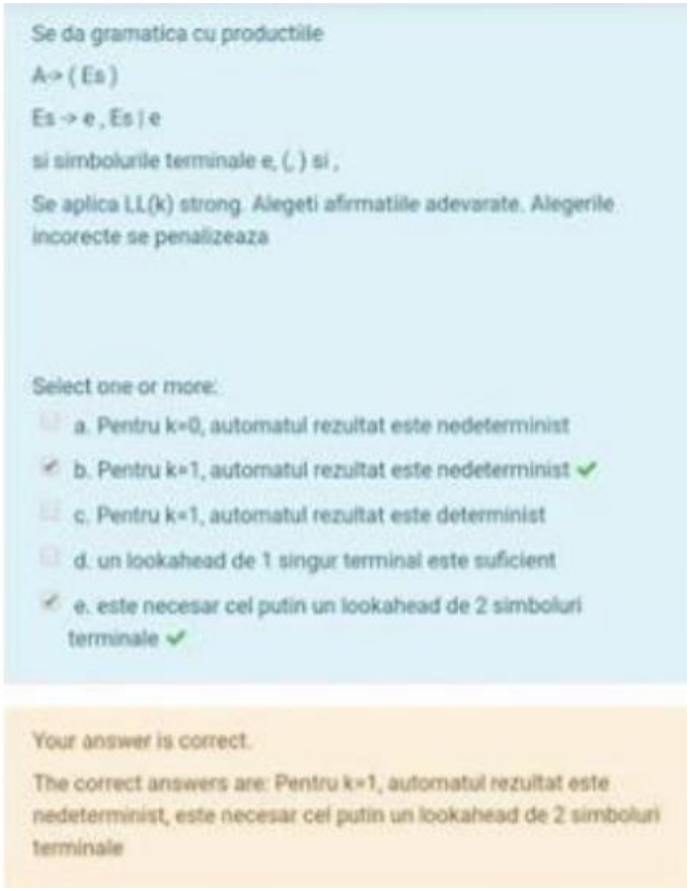
Se da gramatica cu productiile

A -> (Es)

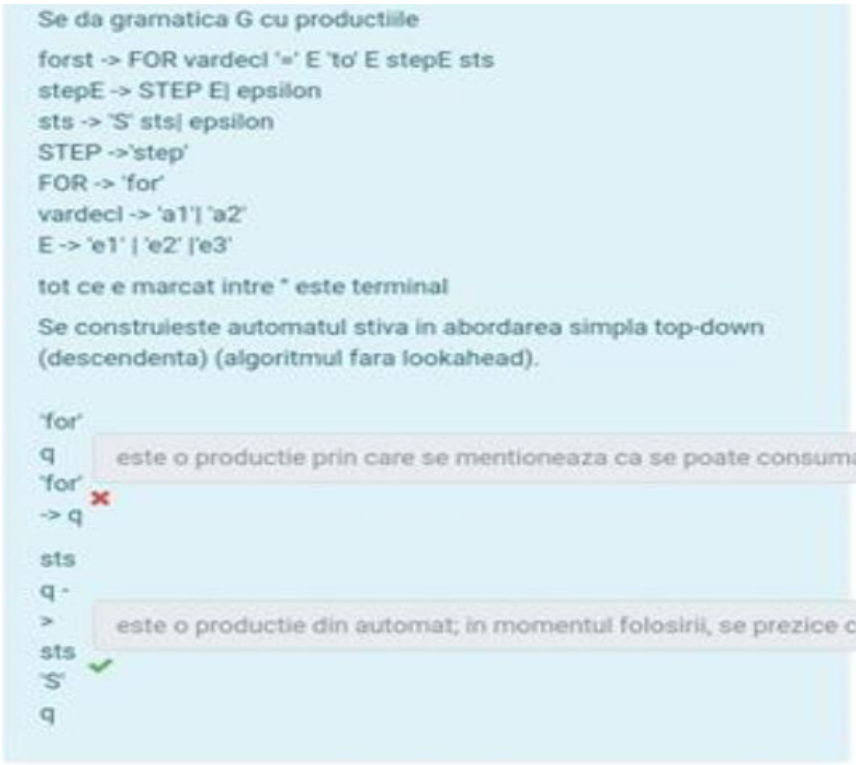
Es -> e, Es | e

Si simbolurile terminale e, (,) si ,

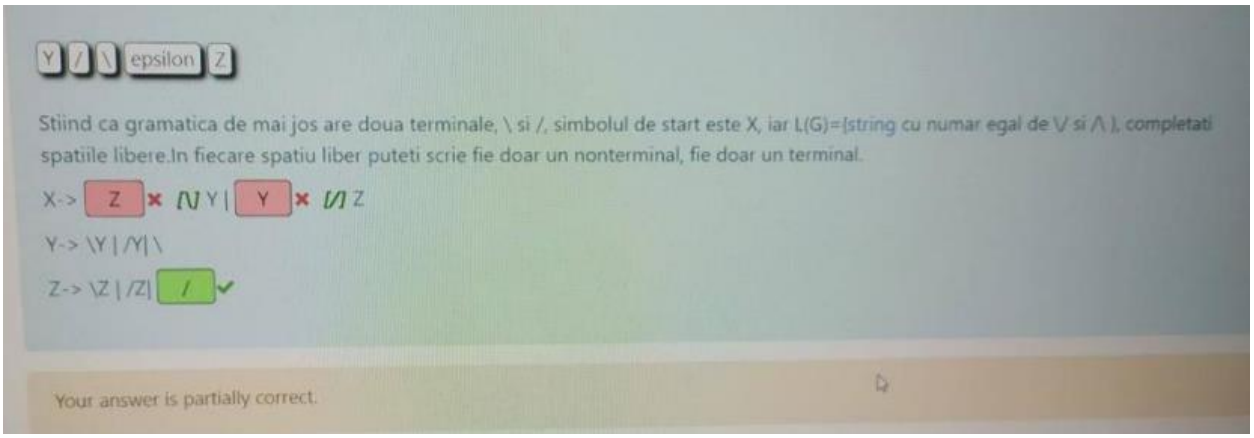
Se aplica LL(k) strong. Alegeti afirmatiile adevarate.



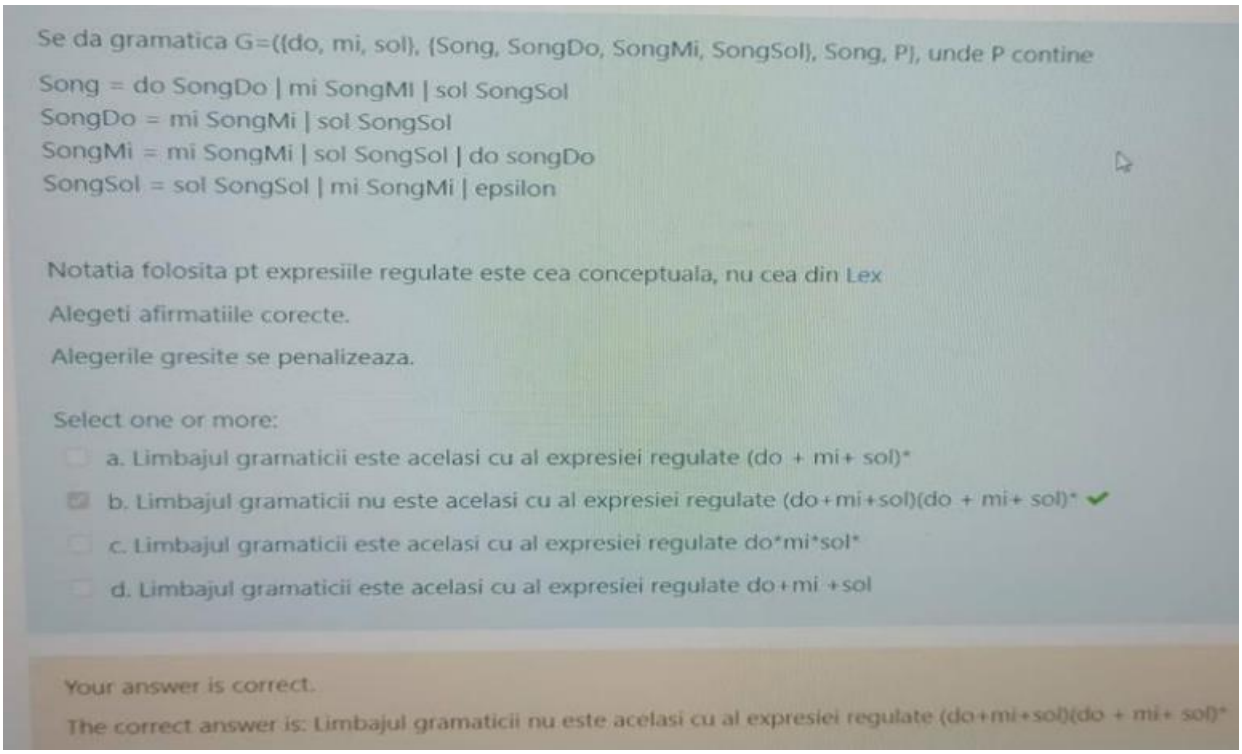
Se da gramatica G cu productiile
forst -> FOR vardecl ‘=’ E ‘to’ stepE sts
stepE -> STEP E| epsilon
sts -> ‘S’ sts| epsilon
STEP -> ‘step’
FOR -> ’for’
Vardecl -> ‘a1’|’a2’
E -> ‘e1’ | ‘e2’ | ‘e3’
Tot ce e marcat intre ‘ este terminal
Se construieste automatul stiva in abordarea simpla top-down



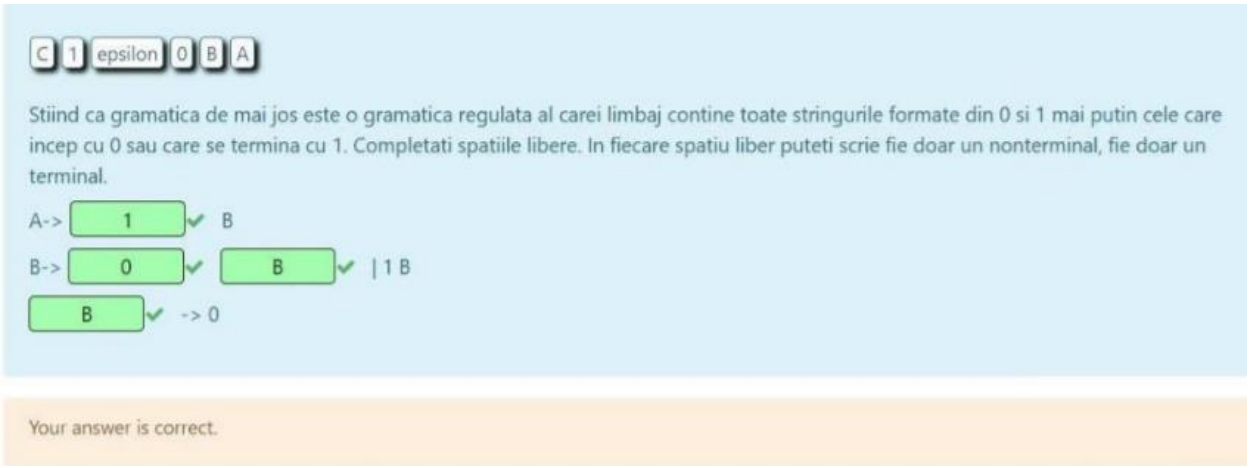
Stiind ca gramatica de mai jos are doua terminale , \ si / , simbolul de start este X , iar L(G) = {string cu numar egal de V si ^ }, completati spatiile libere .



Se da gramatica G=({do, mi, sol}, {Song, SongDo, SongMi, SongSol}, Song, P}, unde P contine
Song = do SongDo | mi SongMi | sol SongSol
SongDo = mi SongMi | sol SongSol
SongMi = mi SongMi | sol SongSol | do SongDo
SongSol = sol SongSol | mi SongMi | epsilon



Stiind ca gramatica de mai jos este o gramatica regulate al carei limbaj contine toate stringurile gormate din 0 si 1 mai putin cele care incep cu 0 sau care se termina cu 1 .



Se dau gramaticile G1 = ({e, ’,’},{P, Ps}, Ps , Productii1}, G2 = ({e, ‘,’}, {P, Ps}, Ps , Productii2},

Se da gramatica cu productiile
A-> (Es)
Es -> e, Es | e
si simbolurile terminale e,(,) si ,
Se aplica LL(2) strong

Se da gramatica cu productiile

A-> (Es)

Es -> e , Es | e

si simbolurile terminale e, (,) si ,

Se aplica LL(2) strong. Alegeti afirmatiile adevarate. Alegerile incorecte se penalizeaza

Select one or more:

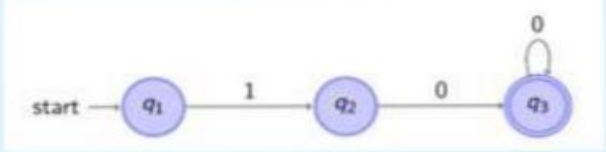
- a. Se porneste de la situatia $q_0 = [A \rightarrow .(Es)]$ ✔
- b. in analiza lui q0 se identifica o noua stare $q_1 = [A \rightarrow .(Es)]$
- c. in analiza lui q0 se identifica o noua stare $q_1 = [Es \rightarrow .e, Es]$
- d. Din starea q_0 se adauga productia $q_0(\rightarrow q_1$ ✔
- e. Din starea q_0 se adauga productia $q_0 e \rightarrow q_1$

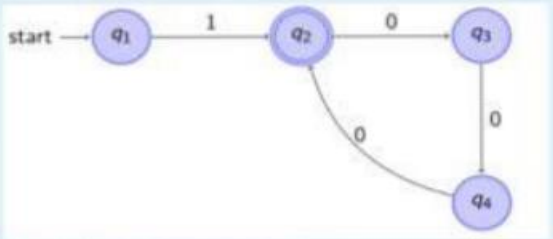
Your answer is partially correct.

You have correctly selected 2.

The correct answers are: Se porneste de la situatia $q_0 = [A \rightarrow .(Es)]$
, in analiza lui q0 se identifica o noua stare $q_1 = [A \rightarrow .(Es)]$
, Din starea q_0 se adauga productia $q_0(\rightarrow q_1$

Se dau urmatoarele 2 automate





Alegeti afirmatiile corecte. Alegeri incorecte se penalizeaza

Select one or more:

- a. orimul are acelasi limbaj cu 10*
- b. au acelasi limbaj
- c. sunt diferite, primul este echivalent cu 100*, in timp ce al doilea accepta si 1 ✔

Se da gramatica G cu productiile

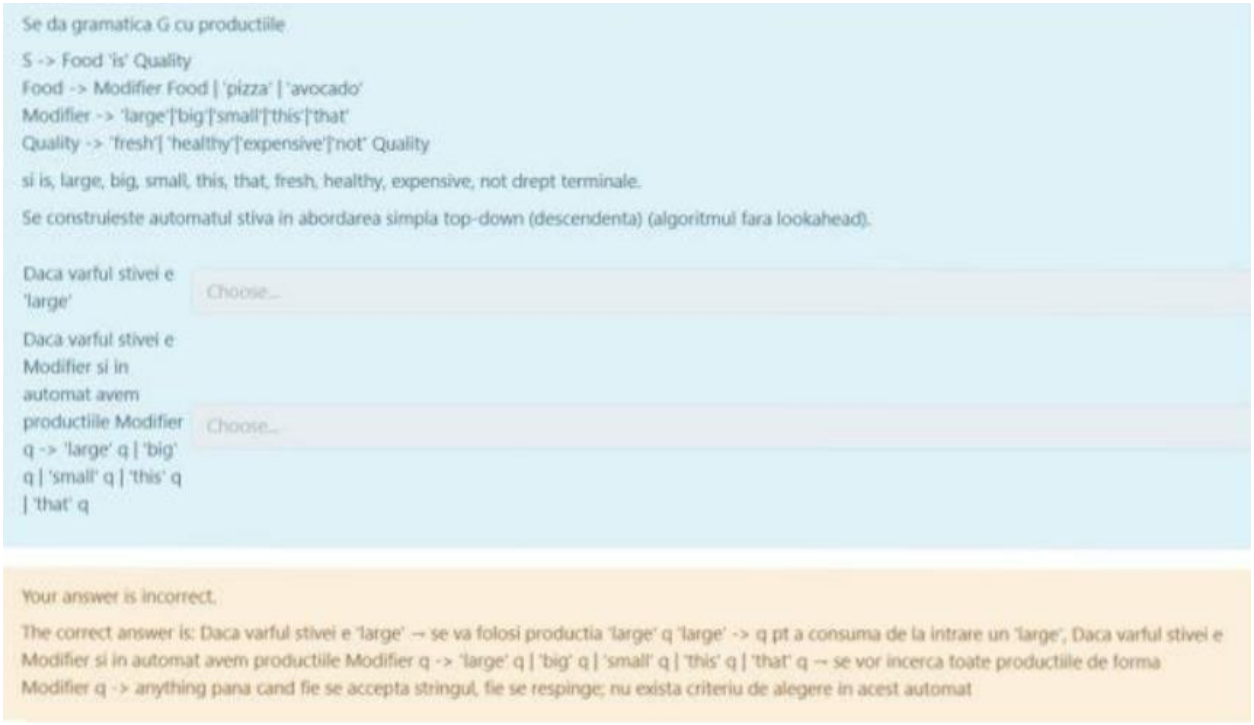
S-> Food ‘is’ Quality

Food -> Modifier Food | ‘pizza’ | ‘avocado’

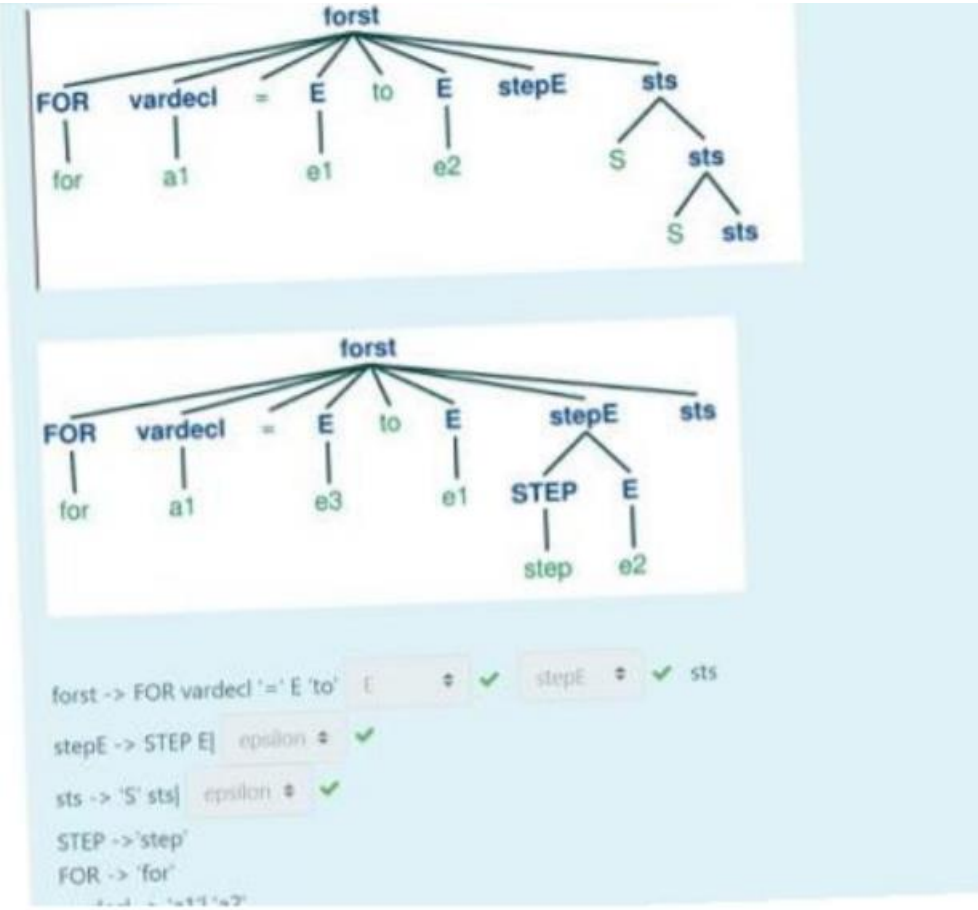
Modifier -> ‘large’ | ‘big’ | ‘small’ | ‘this’ | ’that’

Quality -> ‘fresh’ | ‘healthy’ | ‘expensive’ | ‘not’ Quality

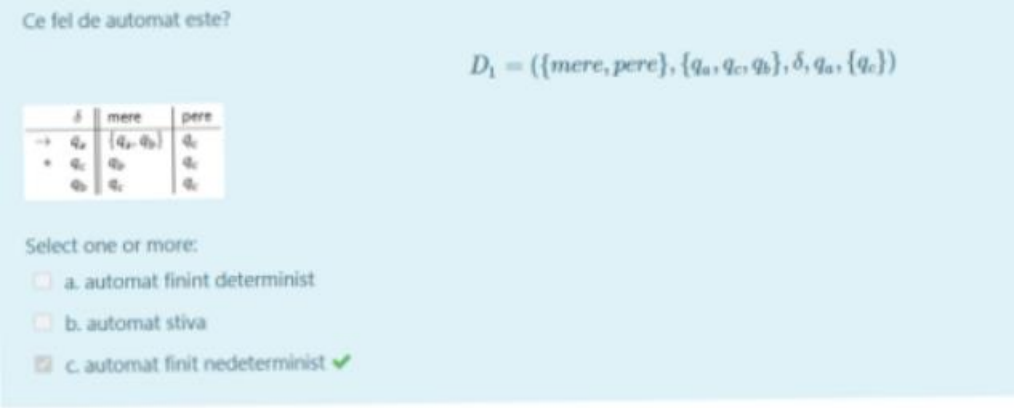
Si is , large, big, small, this, that , fresh , healthy, expensive , not drept terminale .



Completati gramatica



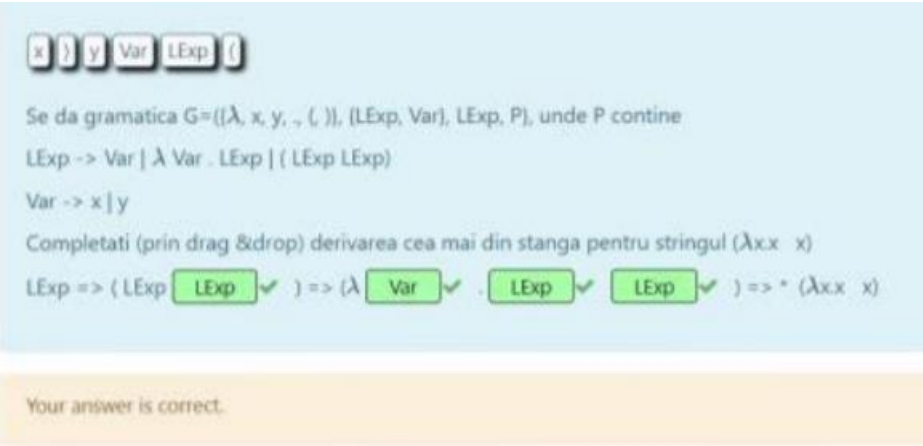
Ce fel de automat este?



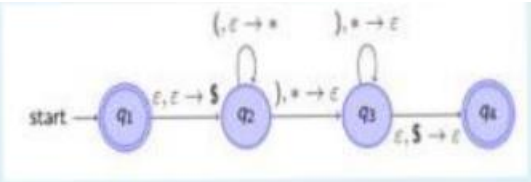
Se da gramatica G=({ λ, x, y, ., (,)}, {LExp, Var} ,LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp)

Var -> x|y



Fie automatul



Fie automatul

$$M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$$

Alegeti raspunsul corect

Select one or more:

- a. este un automat stiva ✔
- b. este un automat finit nedeterminist
- c. este un automat finit determinist

Your answer is correct.

The correct answer is: este un automat stiva

Se da gramatica G cu productiile
forst -> FOR vardecl ‘=’ E ‘to’ stepE sts
stepE -> STEP E| epsilon
sts -> ‘S’ sts| epsilon
STEP -> ‘step’
FOR -> ’for’
Vardecl -> ‘a1’|’a2’
E -> ‘e1’ | ‘e2’ | ‘e3’
Tot ce e marcat intre ‘ este terminal
Se construieste automatul stiva in abordarea simpla top-down

Se da gramatica G cu productiile

forst -> FOR vardecl '=' E 'to' E stepE sts
stepE -> STEP E| epsilon
sts -> 'S' sts| epsilon
STEP -> 'step'
FOR -> 'for'
vardecl -> 'a1'| 'a2'
E -> 'e1' | 'e2' |'e3'

terminalele sunt toate stringurile marcate intre ''

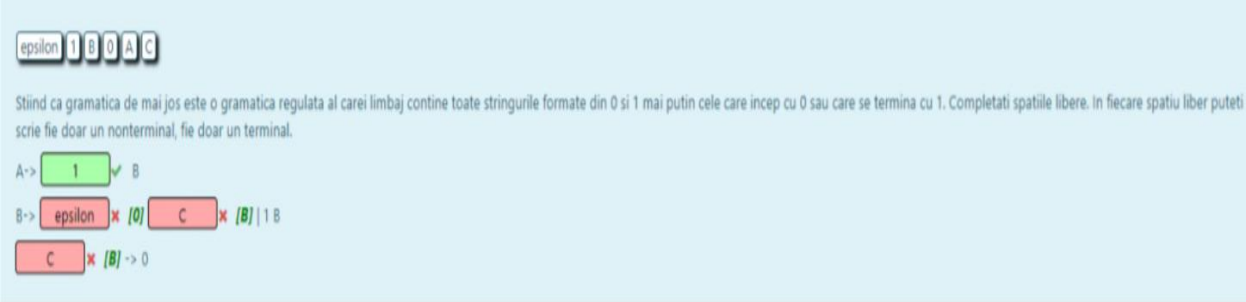
Se construieste automatul stiva in abordarea simpla top-down (descendenta) (algoritmul fara lookahead).

stepE q -> E STEP q	este o productie din acest automat prin care se inlocuieste un simbol de pe stiva cu partea dreapta a unei productii din gramatica in ordinea inversata	✔
'e1' q 'e1' -> q	este o productie din acest automat de tip consum de la intrare pe baza stivei	✔

Your answer is correct.

The correct answer is: stepE q -> E STEP q → este o productie din acest automat prin care se inlocuieste un simbol de pe stiva cu partea dreapta a unei productii din gramatica in ordinea inversata, 'e1' q 'e1' -> q → este o productie din acest automat de tip consum de la intrare pe baza stivei

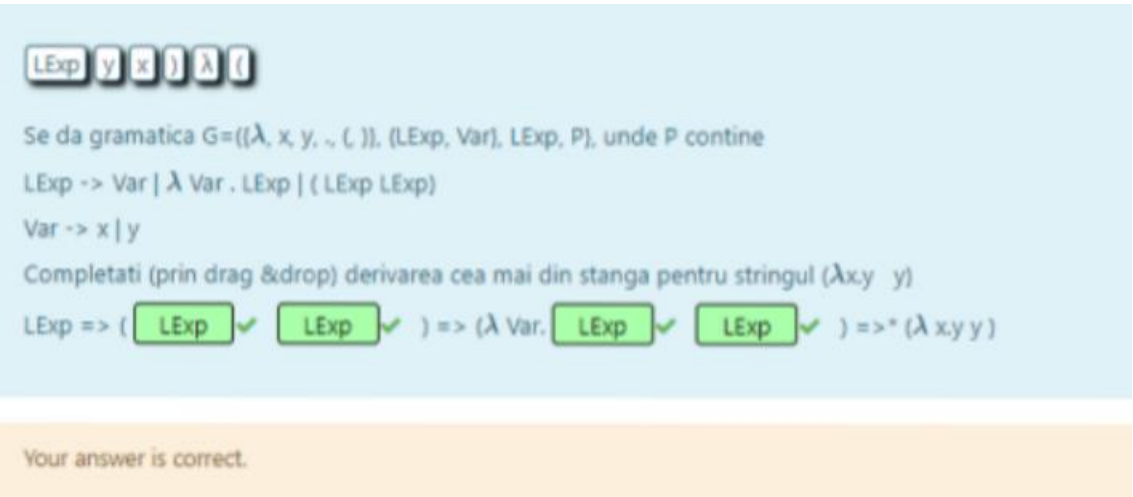
Stiind ca gramatica de mai jos este o gramatica regulate all carei limbaj contine toate stringurile formate din 0 si 1 mai putin cele care incep cu 0 sau care se termina cu 1.



Se da gramatica G=({ λ, x, y, ., (,)}, {LExp, Var} ,LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp)

Var -> x|y

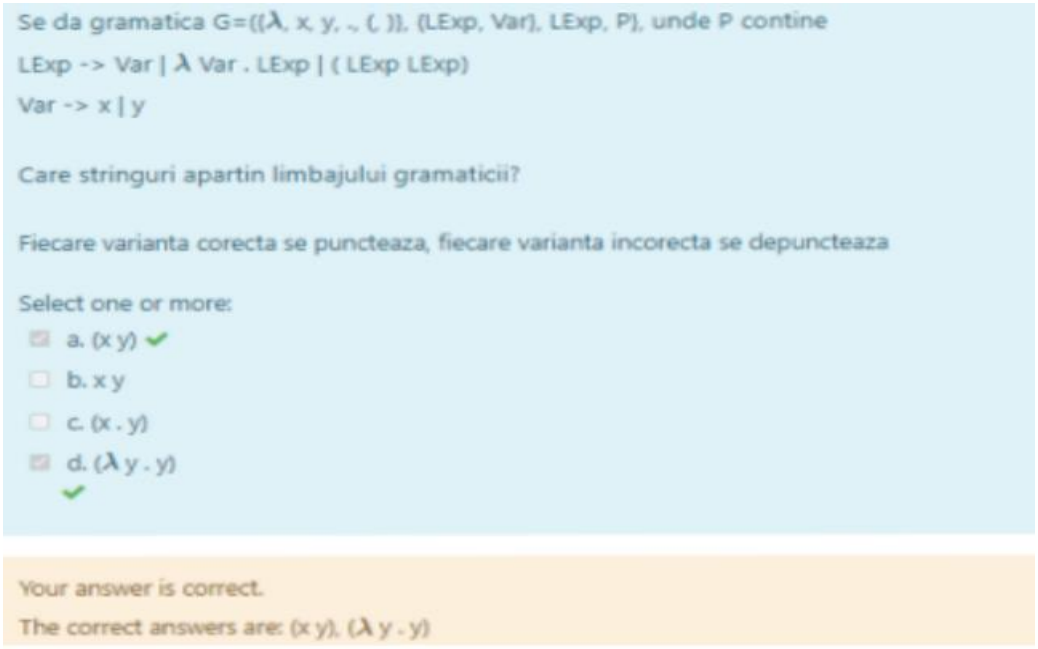


Your answer is correct.

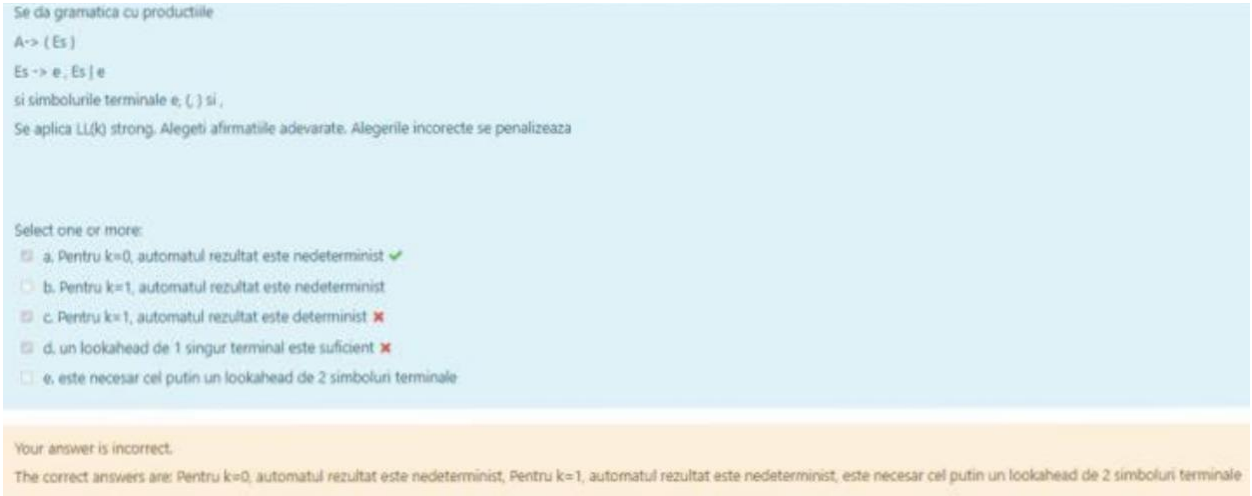
Se da gramatica G={{ λ, x, y, ., (,)}, {LExp, Var} ,LExp, P}, unde P contine

LExp -> Var | λ Var . LExp | (LExp LExp)

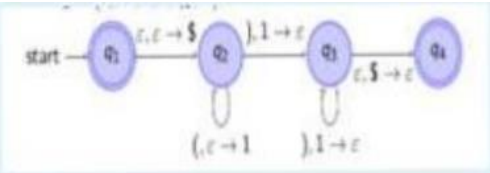
Var -> x|y



Se da gramatica cu productiile
A-> (Es)
Es -> e, Es | e
si simbolurile terminale e,(,) si ,
Se aplica LL(k) strong



Fie automatul



Fie automatul

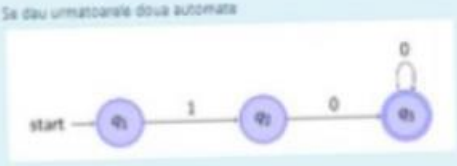
$$M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$$

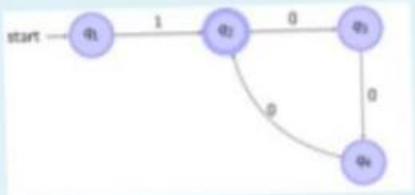
Alegeti raspunsul corect

Select one or more:

- a. in starea q2 citeste) doar daca poate scoate de pe stiva un 1, altfel revine in q1
- b. in starea q2 citeste) doar daca poate scoate de pe stiva un 1, altfel stringul e respins ✔
- c. in starea q2 citeste) doar daca poate scoate de pe stiva (

Se dau urmatoarele doua automate





Alegeti afirmatiile corecte. Alegeri incorecte se penalizeaza

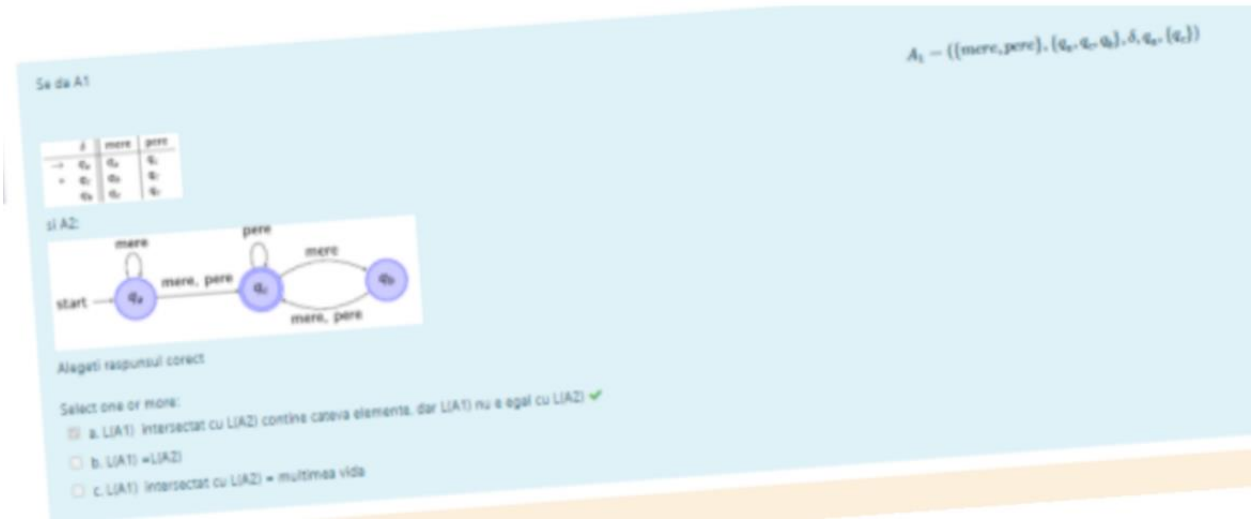
Select one or more:

- a. primul are acelasi limbaj cu 10*
- b. au acelasi limbaj
- c. sunt diferite, primul este echivalent cu 100*, in timp ce al doilea accepta si 1 ✔

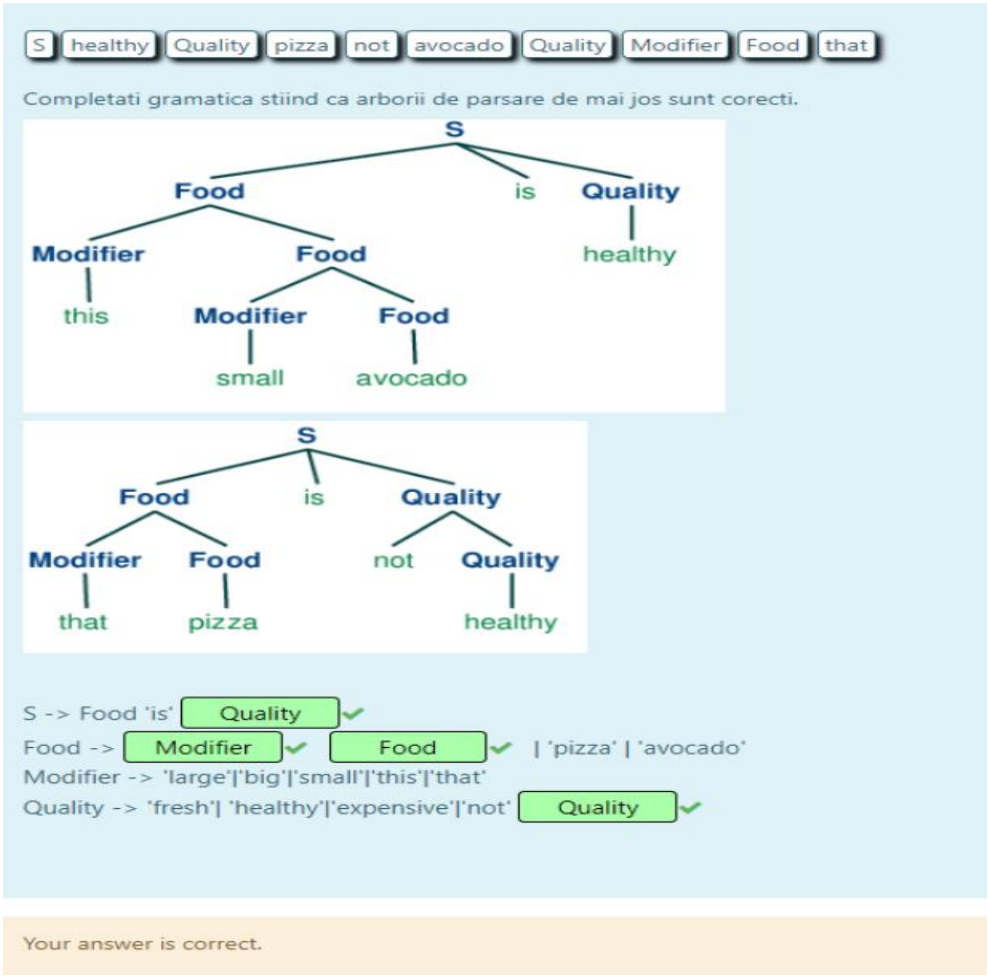
Your answer is correct.

The correct answer is: sunt diferite, primul este echivalent cu 100*, in timp ce al doilea accepta si 1

Se da A1



Completati gramatica stiind ca arborii de parsare de mai jos sunt corecti



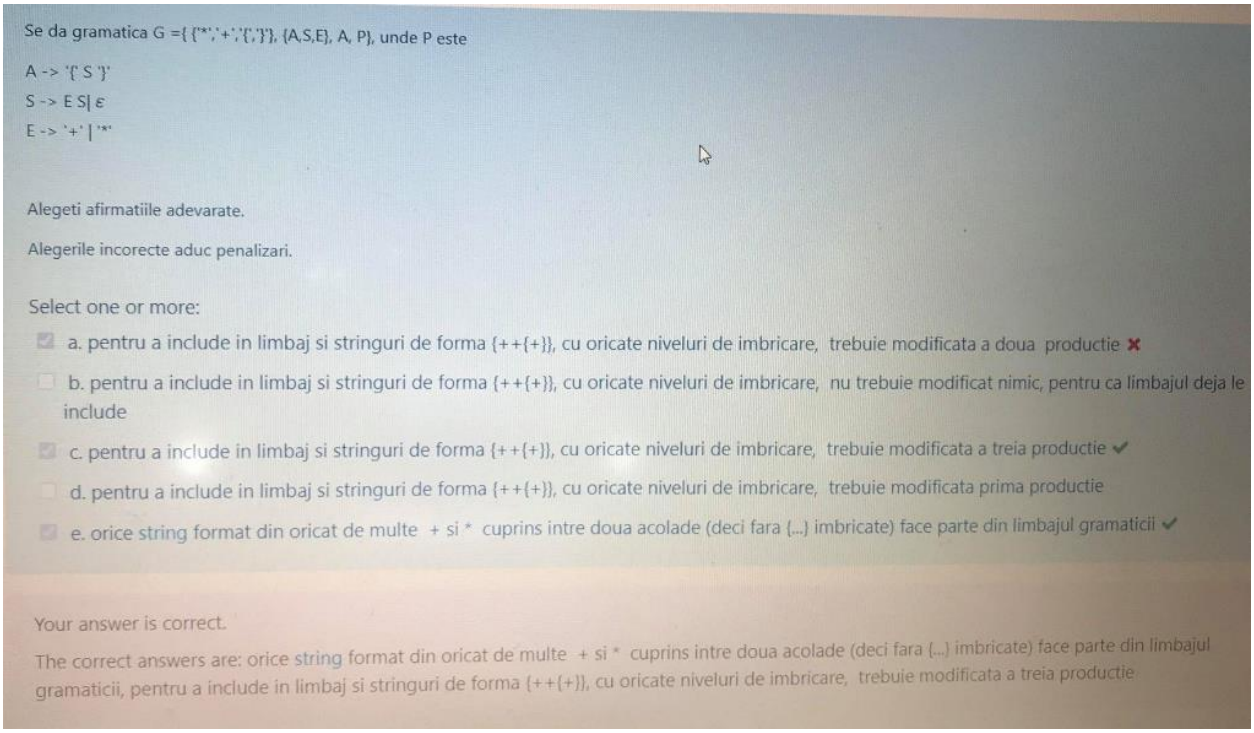
Your answer is correct.

Se da gramatica G ={ {‘*’. ‘+’,’{‘,’}’}, {A,S,E}, A,P}, unde P este

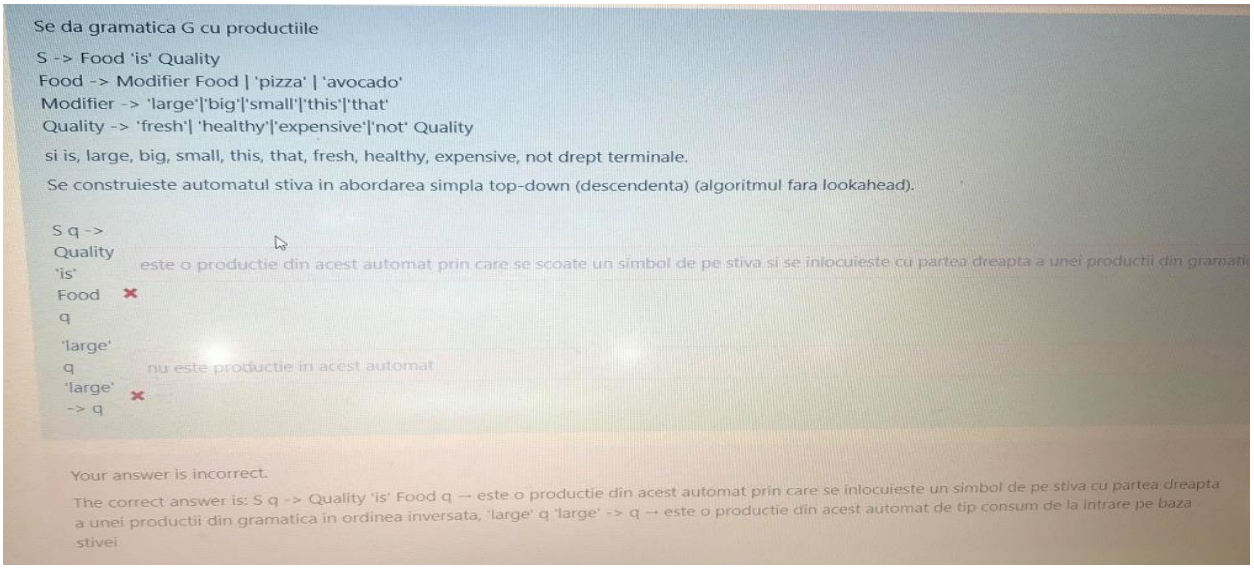
A -> ‘{‘ S ‘}’

S -> E S|epsilon

E -> ‘+’ | ‘*’



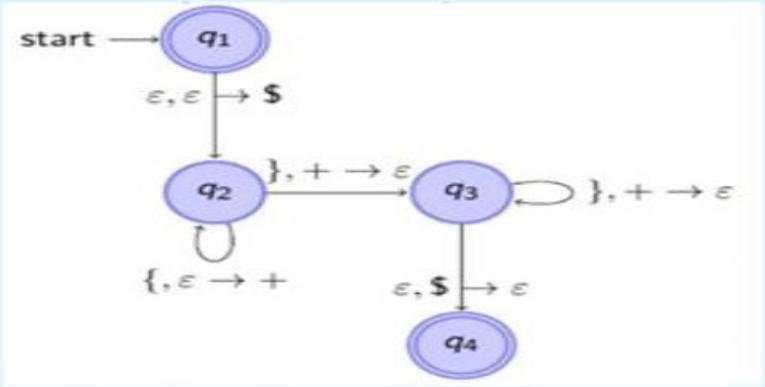
Se da gramatica G cu productiile
S-> Food ‘is’ Quality
Food -> Modifier Food | ‘pizza’ | ‘avocado’
Modifier -> ‘large’ | ‘big’ | ‘small’ | ‘this’ | ’that’
Quality -> ‘fresh’ | ‘healthy’ | ‘expensive’ | ‘not’ Quality
Si is , large, big, small, this, that , fresh , healthy, expensive , not drept terminale .



Fie automatul

Fie automatul

$$M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$$



Alegeti raspunsul corect

Select one or more:

- a. automatul accepta doar stringuri care contin n acolade deschise urmate de n acolade inchise ✔
- b. automatul accepta stringul { { } { } }
- c. automatul accepta doar stringuri care contin n acolade deschise urmate de m acolade inchise, unde m si n pot fi diferite

Se dau gramaticile G1=({e, ‘,’}, {P,Ps}, P2, Productii1}, G2=({e, ‘,’}, {P,P2},Ps, Productii2},

Se dau gramaticile G1=({e, ','}, {P, Ps}, Ps, Productii1}, G2=({e, ','}, {P, Ps}, Ps, Productii2},

Productii1 = { Ps-> P ',' Ps | ε

P-> e}

Productii2 = { Ps-> P ',' Ps | P

P-> e}

Care afirmatii sunt adevarate?

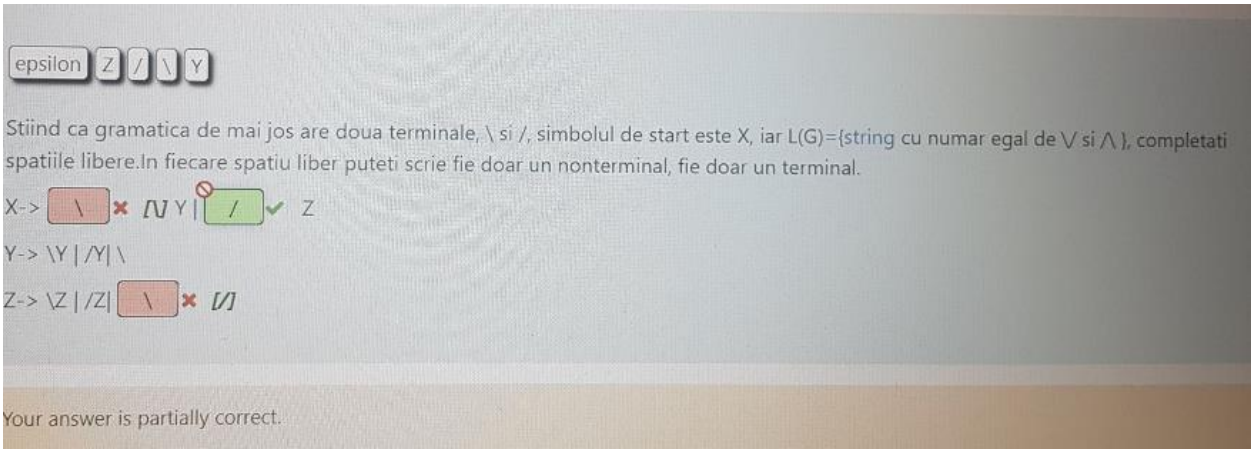
e,e,e apartine L(G2), dar nu si L(G1) ✔

e,e,e, apartine L(G1), dar nu si L(G2) ✔

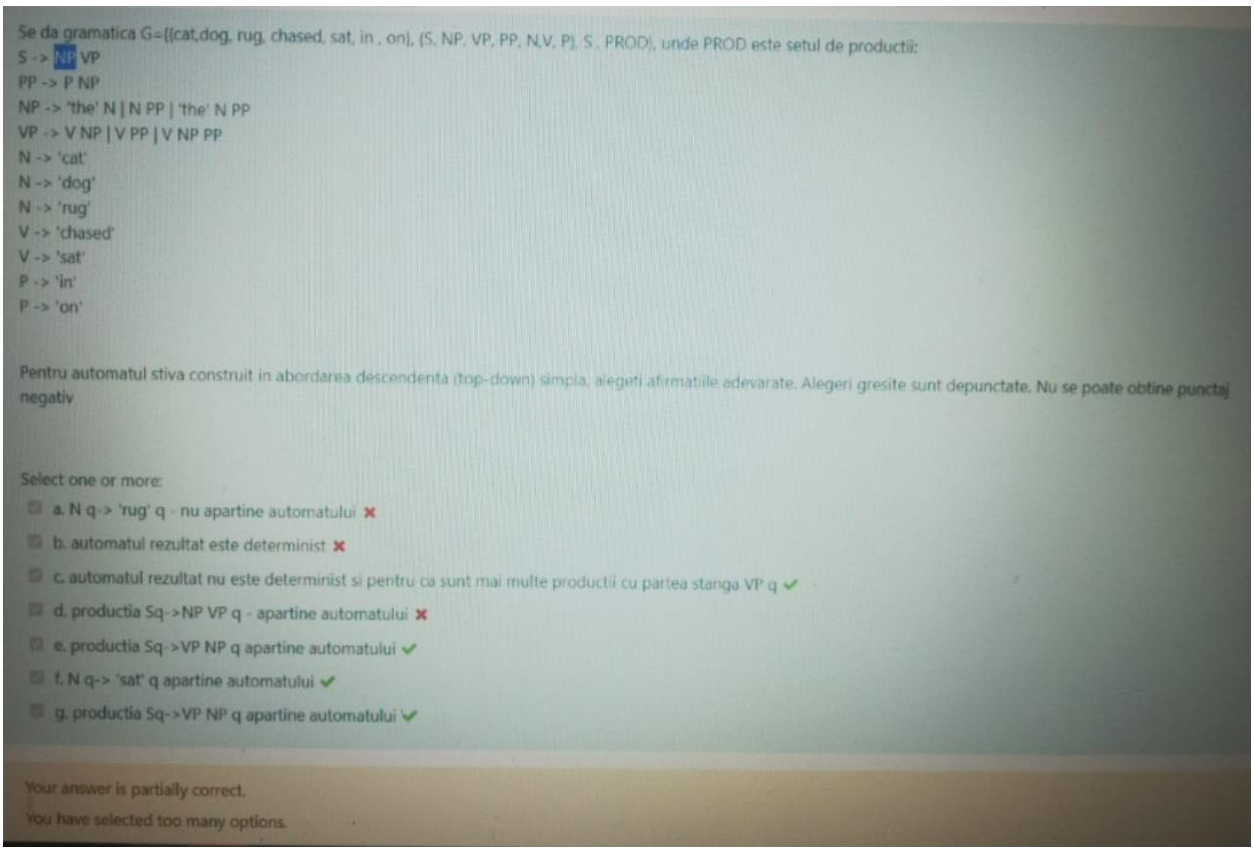
Your answer is correct.

The correct answer is: e,e,e apartine → L(G2), dar nu si L(G1), e,e,e, apartine → L(G1), dar nu si L(G2)

Stiind ca gramatica de mai jos are doua terminale , \ si / , simbolul de start este X, iar L(G) = {strin cu numar egal de V si ^ }, completati spatiile libere.



Se da gramatica G={cat,dog,rug,chased,sat,in,on}, {S,NP,VP,PP,N,V,P},S, PROD}, unde PROD este setul de productii:



Se da gramatica G cu productiile

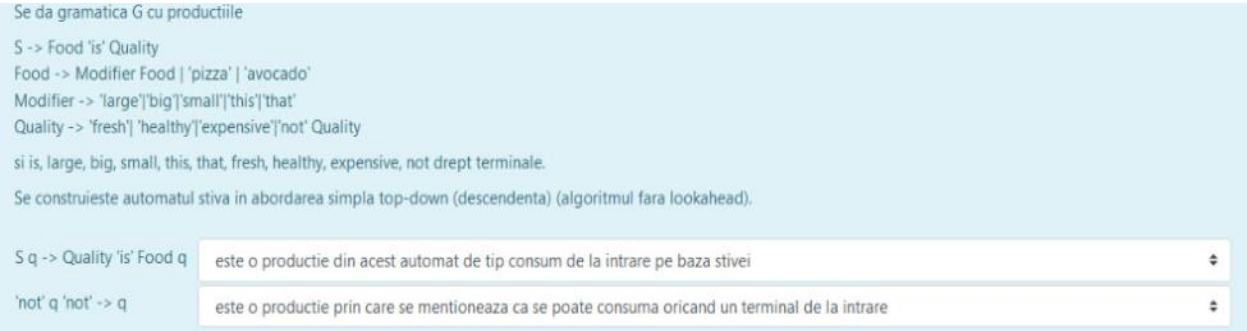
S-> Food ‘is’ Quality

Food -> Modifier Food | ‘pizza’ | ‘avocado’

Modifier -> ‘large’ | ‘big’ | ‘small’ | ‘this’ | ’that’

Quality -> ‘fresh’ | ‘healthy’ | ‘expensive’ | ‘not’ Quality

Si is , large, big, small, this, that , fresh , healthy, expensive , not drept terminale .



Stiind ca gramatica de mai jos are terminalele { } 1 si 2 , siimbolul de start este A, iar arborele de mai jos este arborele de parsare pentru { 1 { 2 } 2 } , completati gramatica