

Технически университет - Варна

А. Антонов, Вл. Николов, Ив. Пенев

ДИСКРЕТНИ СТРУКТУРИ

Ръководство за семинарни
упражнения

Варна

2015

УДК 519.21: 519.87 (076)

Настоящото ръководство е предназначено за студентите от специалности „Компютърни системи и технологии“ и „Софтуерни и интернет технологии“ в Технически университет - Варна. Съдържанието на ръководството е съобразено с учебната програма по дисциплината "Дискретни структури".

© Анатолий Стефанов Антонов
Владимир Николов Николов
Ивайло Пламенов Пенев

ISBN

ПРЕДГОВОР

Ръководството за семинарни упражнения по Дискретни структури е предназначено за студентите от специалности „Компютърни системи и технологии“ и „Софтуерни и интернет технологии“ в Технически Университет-Варна. Ръководството е съобразено с приетата учебна програма по едноименната дисциплина „Дискретни структури“, включена като задължителна в държавните изисквания за специалността. Изложеният материал може да бъде използван за подготовка по дисциплината и от студенти по същата специалност в други технически университети или от студенти, обучавани в сродни специалности.

Основната цел на ръководството е да затвърди знанията на студентите, получени от лекционния материал, чрез разглеждане на примери и самостоятелно решаване на задачи, свързани с моделиране на поведението на дискретни системи, по време на семинарните упражнения. Ръководството е структурирано в 15 теми, които съответстват на лекционния материал по дисциплината. Подходите за моделиране и за решаване на задачи са представени накратко в уводната част на всяка тема, като след това те са онагледени и с подходящи и подробно разработени примери. Следните по-важни теми формират съдържанието на ръководството:

- Множества, отношения и функции върху множества. Регулярни множества и регулярни изрази.
- Формални граматики и езици. Синтактични графи.
- Абстрактни автомати. Машина на Тюринг. Крайни и стекови автомати. Базови и частично рекурсивни функции.
- Паралелни системи и процеси. Мрежи на Петри. Мрежови езици. Координиране и синхронизиране на паралелни процеси.
- Абстрактни информационни структури. Представяне на модели.
- Математическа логика. Съждителна логика. Предикатна логика. Машина за логически извод. Резолюция и унификация.

Упражнение 1.

Множества. Операции с множества.

Теоретичен увод

Определения:

Празно множество: $\emptyset$ множество без елементи.

Универсално множество: U

Отношение принадлежност: $x \in A$ - x принадлежи на множеството A .

Отношение непринадлежност: $x \notin A$ - x не принадлежи на множеството A .

Равенство на множества: $A = B$ - елементите на множествата A и B съвпадат.

Неравенство на множества: $A \neq B$ - елементите на множествата A и B не съвпадат.

Подмножество: $A \subseteq B$ - A е подмножество на B ако всеки елемент на A е елемент на B или двете множества са равни т.е.,

$$A \subset B; A = B.$$

Собствено подмножество: $A \subset B$ - A е собствено подмножество на B ако всеки елемент на A е елемент на B т.е.,

$$A \subseteq B, A \neq B.$$

Операции с множества. За две множества A и B са дефинирани следните операции:

Обединение на множества: $A \cup B$

$$A \cup B = \{x \mid x \in A; x \in B\}.$$

Сечение на множества: $A \cap B$

$$A \cap B = \{x \mid x \in A; x \in B\}.$$

Разлика на множества: $A \setminus B$

$$A \setminus B = \{x \mid x \in A, x \notin B\}.$$

Симетрична разлика на множества: $A \div B$

$$(A \setminus B) \cup (B \setminus A)$$

Допълнение на множество: $\overline{A}$

$x \in U$ и $x \notin A$, където

U - универсално множество. Множеството на всички елементи от предметната област.

Примерни задачи:

Задача 1. Да се докаже равенството:

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

Решение:

I. Лява част. Доказва се подмножеството:

$$A \cap (B \cup C) \subseteq (A \cap B) \cup (A \cap C)$$

Нека

$$x \in A \cap (B \cup C)$$

$$\Rightarrow x \in A, x \in (B \cup C).$$

Щом $x \in (B \cup C) \Rightarrow x \in B; x \in C$

1. Допуска се, че $x \in B$ и $x \in A$

$$\Rightarrow x \in A \cap B \Rightarrow x \in (A \cap B) \cup (A \cap C), \text{ тъй като всеки}$$

елемент принадлежащ на едно множество $(A \cap B)$, принадлежи и на обединението на това множество с всяко друго множество $(A \cap C)$.

2. Допуска се, че $x \in C$ и $x \in A$

$$x \in C, x \in A \Rightarrow x \in (A \cap C) \Rightarrow$$

$$x \in (A \cap C) \cup (A \cap B) \Rightarrow x \in (A \cap B) \cup (A \cap C),$$

поради комутативност на операцията обединение на множества.

$$\text{От 1 и 2} \quad \Rightarrow x \in (A \cap B) \cup (A \cap C)$$

II. Дясна част. Доказва се подмножеството:

$$A \cap (B \cup C) \supseteq (A \cap B) \cup (A \cap C).$$

Нека

$$x \in (A \cap B) \cup (A \cap C) \Rightarrow x \in A \cap B; x \in A \cap C.$$

1. Допуска се, че

$$x \in (A \cap B) \Rightarrow x \in A, x \in B \Rightarrow x \in A, x \in B \cup C,$$

защото всеки елемент, принадлежащ към дадено множество, принадлежи и на обединението на това множество с кое да е друго - (C)

$$\Rightarrow x \in A \cap (B \cup C).$$

2. Допуска се, че

$$x \in A \cap C \Rightarrow x \in A, x \in C \Rightarrow$$

$$\Rightarrow x \in A, x \in C \cup B \Rightarrow x \in A, x \in B \cup C,$$

поради дистрибутивност на операция обединение на множества.

От 1 и 2

$$\Rightarrow x \in A \cap (B \cup C),$$

т. е.

$$(A \cap B) \cup (A \cap C) \subseteq A \cap (B \cup C)$$

от I и II следва равенството на множествата.

Задача 2. Да се докаже равенството:

$$\overline{A \cap B} = \overline{A} \cup \overline{B}$$

Решение:

I. Нека

$$x \in \overline{A \cap B} \Rightarrow x \in U, x \notin A \cap B$$

Щом

$$x \notin A \cap B \Rightarrow x \notin A; x \notin B$$

1. Нека

$$x \notin A \Rightarrow x \in \overline{A} \Rightarrow x \in \overline{A} \cup \overline{B}$$

2. Нека

$$x \notin B \Rightarrow x \in \overline{B} \Rightarrow x \in \overline{A} \cup \overline{B}$$

От 1 и 2 следва:

$$\Rightarrow x \in \overline{A} \cup \overline{B},$$

т.е.:

$$\Rightarrow \overline{A \cap B} \subseteq \overline{A} \cup \overline{B}$$

II. Нека

$$x \in \overline{A} \cup \overline{B} \Rightarrow x \in \overline{A}; x \in \overline{B}$$

1. Ако

$$x \in \overline{A} \Rightarrow x \in U \text{ и}$$

$$x \notin A \Rightarrow x \notin A \cap B \Rightarrow x \in \overline{A \cap B}$$

2. Ако

$$x \in \overline{B} \Rightarrow x \in U \text{ и}$$

$$x \notin B \Rightarrow x \notin B \cap A \Rightarrow x \notin A \cap B \Rightarrow x \in \overline{A \cap B}$$

От 1 и 2 следва:

$$\Rightarrow \overline{A} \cup \overline{B} \subseteq \overline{A \cap B}$$

От I и II следва равенството на множествата:

$$\Rightarrow \overline{A \cap B} = \overline{A} \cup \overline{B}$$

Задача 3. Да се докаже равенството:

$$A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$$

Решение:

I. Нека $A \setminus (B \cup C) \Rightarrow x \in A, x \notin B \cup C \Rightarrow$

$$\Rightarrow x \in A, x \in \overline{B}, x \in \overline{C}$$

$$\Rightarrow x \in A \setminus B, x \in A \setminus C$$

II. Нека

$$x \in (A \setminus B) \cap (A \setminus C)$$

$$\Rightarrow x \in (A \setminus B), x \in (A \setminus C) \Rightarrow x \in A, x \notin B, x \notin C$$

$$\Rightarrow x \in A, x \notin B \cup C \Rightarrow x \in A, x \in \overline{B \cup C}$$

От I и II следва равенството на двете множества.

$$\Rightarrow A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$$

Задача 4. Да се докаже, че:

$$A \cap B \subseteq C \Leftrightarrow A \subseteq \overline{B} \cup C$$

Решение:

I. Допуска се

$$A \cap B \subseteq C \quad (1)$$

Нека $x \in A$, за x има две възможности:

1. $x \in A$ и $x \in B$, тогава от (1), следва че $x \in C$. Щом

$x \in C$, тогава $x \in C \cup \overline{B} \Rightarrow x \in \overline{B} \cup C$;

2. $x \in A$ и $x \notin B$

$$\Rightarrow x \in A, x \in \overline{B} \Rightarrow x \in \overline{B} \cup C$$

От 1 и 2

$$\Rightarrow A \subseteq \overline{B} \cup C$$

II. Допуска се

$$A \subseteq \overline{B} \cup C \quad (2)$$

Нека $x \in A \cap B$

$$\Rightarrow x \in A, x \in B.$$

От (2)

$$\Rightarrow x \in C, \text{ тъй като } A \subseteq \overline{B} \cup C \text{ и } x \in B \Rightarrow x \in C$$

има противоречие с $x \in B$, т.е.

$$\Rightarrow A \cap B \subseteq C$$

От I и II следва еквивалентността на двете подмножества:

$$\Rightarrow A \cap B \subseteq C \Leftrightarrow A \subseteq \overline{B} \cup C$$

Задача 5. Да се докаже тъждеството:

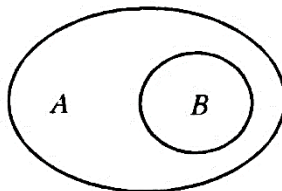
$$(A \setminus B) \cup B = A \Leftrightarrow B \subseteq A$$

Решение:

Нека

$$(A \setminus B) \cup B = A \text{ и } x \in B \Rightarrow x \in A$$

Отношението на множествата А и В, представени чрез диаграма на Вен е следното:



Фиг. 1.1. Диаграма на Вен

За този случай:

$$\begin{aligned}(A \setminus B) \cup B &= (A \cap \overline{B}) \cup B = (A \cup B) \cap (B \cup \overline{B}) = \\ &= (A \cup B) \cap U = A\end{aligned}$$

Примерни задачи за упражнение:

Да се докаже:

а) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

б) $(A \cup B) \cap A = (A \cap B) \cup A = A$

в) $A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$

г) $A \setminus (A \setminus B) = A \cap B$

д) $A \setminus B = A \setminus (A \cap B)$

е) $A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C) = (A \cap B) \setminus C$

ж) $(A \cap B) \cup C = A \cap (B \cup C) \Leftrightarrow C \subseteq A$

Упражнение 2.

Декартово произведение на множества.

Отношения и функции в множествата.

Теоретичен увод

Определения:

Неподредена n -торка от елементи:

Множество от обекти $B = \{b_1, b_2, \dots, b_n\}$, елементи на което се явяват $b_1, b_2, \dots, b_n$ и само те, се нарича неподредена n -торка от елементи. $C = \{x, y\}$, където $x \neq y$, се наричат неподредена двойка (двойка) обекти x и y . Множеството $\{x, x\}$ се обозначава с $\{x\}$ и се нарича едноелементно множество.

Подредена n -торка от елементи:

Дадени са обекти $b_1, b_2, \dots, b_n$.

$\langle b_1, b_2, \dots, b_n \rangle$ се нарича подредена n -торка от обекти, ако притежават следното основно свойство:

Множеството $\langle b_1, b_2, \dots, b_n \rangle = \langle c_1, c_2, \dots, c_n \rangle$, тогава и само тогава, когато $b_1 = c_1, b_2 = c_2, \dots, b_n = c_n$.

$\langle b_1, b_2 \rangle$ - се нарича подредена двойка, ако $\langle b_1, b_2 \rangle = \langle b_2, b_1 \rangle$ тогава и само тогава, когато $b_1 = b_2$.

Декартово произведение на множества.

$A_1 \times A_2 \times \dots \times A_n = \{ \langle a_1, a_2, \dots, a_n \rangle \mid a_1 \in A_1, a_2 \in A_2, \dots, a_n \in A_n \}$ се нарича декартово произведение на множествата $A_1, A_2, \dots, A_n$.

Нека $Y = \{y \mid y \in Y\}$, $Z = \{z \mid z \in Z\}$ под декартово произведение на двете множества се разбира $Y \times Z = \{ \langle y, z \rangle \mid y \in Y, z \in Z \}$.

Отношения:

n -местно отношение (отношение с n аргумента) на множеството X се нарича всяко подмножество на множеството

на себе си.

2-местно (бинарно) отношение

$$B = \{ \langle x, y \rangle \mid x \in D, y \in R \},$$

където

$$\begin{aligned} D(r) &= \{ x \mid x \in D \} && \text{- множество на аргументи} \\ R(r) &= \{ y \mid y \in R \} && \text{- множество на стойностите.} \end{aligned}$$

1-местно отношение (свойство)

$$A = \{ x \mid x \in D, x \text{ изпълнява условията на свойството} \}.$$

Видове отношения:

Рефлексивно отношение r_p

$$\text{Ако } x \in D(r_p), \text{ то } \langle x, x \rangle \in r_p.$$

Симетрично отношение r_c

$$\text{Ако } \langle x, y \rangle \in r_c, \text{ то } \langle y, x \rangle \in r_c.$$

Транзитивно r_T

$$\text{Ако } \langle x, y \rangle \in r_T \text{ и } \langle y, z \rangle \in r_T, \text{ то } \langle x, z \rangle \in r_T.$$

Обратно r^{-1}

$$r^{-1} = \{ \langle y, x \rangle \mid \langle x, y \rangle \in r \}.$$

Операции в отношенията

Произведение на отношения R_1, R_2

Нека $R_1 \subseteq A \times B$ и $R_2 \subseteq B \times C$

$R_1 \cdot R_2 = \{ \langle x, y \rangle \mid \text{съществува } z \text{ такава, че}$

$$\langle x, z \rangle \in R_1, \langle z, y \rangle \in R_2 \}$$

Функция: Бинарното отношение f се нарича функция ако от $\langle x, y \rangle \in f$ и $\langle x, z \rangle \in f$ следва $y = z$, т.е. за всяка x , принадлежащо на дефиниционната област на функцията f , съществува единствен елемент y , такъв, че $\langle x, y \rangle \in f$. Този елемент y се обозначава с $f(x)$.

Ако $D(f) = X$

$R(f) = Y$

f изобразява X в (на) Y .

Класификация на функциите:

Тъждествена	$f = \{ \langle x, y \rangle \mid x = y, x \in D(f) \} \quad f(x) = x;$
Обратна	$f^{-1}(f(x)) = f(f^{-1}(x));$
Константна	$f = \{ \langle x, y \rangle \mid x \in D(f), y = \{const\} \},$ $f(x) = const;$
Вложена	$f = \{ \langle x, y \rangle \mid x \in D(f), y = h(g(x)) \};$ $f = h \circ g;$
Предикатна	$f = \{ \langle x, y \rangle \mid x \in D(f),$ $y = \{TRUE, FALSE\} \};$
Рекурсивна	$f = \{ \langle x, y \rangle \mid x \in D(f), y = f(x) \}.$

Примерни задачи:

Задача 1. Да се докаже равенството:

$$(A \cup B) \times C = (A \times C) \cup (B \times C)$$

Доказателство: Следва да се докаже, че произволно избран елемент от лявата част на равенството принадлежи на дясната и обратното.

I. За лявата част:

$$x \in (A \cup B) \times C \Rightarrow x = \langle y, z \rangle,$$

където $y \in A \cup B, z \in C,$

$$\text{Ако } y \in A \cup B \Rightarrow y \in A; y \in B$$

Следователно $\langle y, z \rangle \in A \times C; \langle y, z \rangle \in B \times C$

$$\Rightarrow x = \langle y, z \rangle \in (A \times C) \cup (B \times C)$$

II. За дясната част:

$$x \in (A \times C) \cup (B \times C) \Rightarrow x \in A \times C; x \in B \times C;$$

1. Ако $x \in A \times C \Rightarrow x = \langle y, z \rangle$, където $y \in A, z \in C.$

2. Ако $x \in B \times C \Rightarrow x = \langle y, z \rangle$, където $y \in B, z \in C.$

Следователно $y \in (A \cup B), z \in C \Rightarrow x = \langle y, z \rangle \in (A \cup B) \times C.$

Задача 2. Да се докаже равенството:

$$A \times (B \cup C) = (A \times B) \cup (A \times C)$$

I. Лява част:

$$x \in A \times (B \cup C) \Rightarrow x = \langle y, z \rangle, \text{ където } y \in A, z \in B \cup C$$

От $z \in B \cup C \Rightarrow z \in B ; z \in C$

1. $y \in A, z \in B$, то $x = \langle y, z \rangle \in A \times B$

2. $y \in A, z \in C$, то $x = \langle y, z \rangle \in A \times C$
 $\Rightarrow \langle y, z \rangle \in (A \times B) \cup (A \times C)$

II. Дясна част.

$x \in (A \times B) \cup (A \times C) \Rightarrow x \in A \times B ; x \in A \times C$

1. $x \in A \times B \Rightarrow x = \langle y, z \rangle$, където $y \in A, z \in B$

2. $x \in A \times C \Rightarrow x = \langle y, z \rangle$, където $y \in A, z \in C$
 $\Rightarrow \langle y, z \rangle \in A \times (B \cup C)$

Задача 3. Да се докаже, че:

$$(A \setminus B) \times C = (A \times C) \setminus (B \times C)$$

I. Лява част.

$x \in (A \setminus B) \times C \Rightarrow x = \langle y, z \rangle$, където $y \in A \cap \overline{B}, z \in C$

$y \in A, z \in C \Rightarrow x \in A \times C$

$y \in \overline{B}, z \in C \Rightarrow x \in \overline{B} \times C$

$x \in (A \times C) \setminus (B \times C)$ принадлежи на дясното множество.

II. Дясна част:

$x \in (A \times C) \setminus (B \times C) \Rightarrow x \in A \times C, x \notin B \times C;$

$\Rightarrow x \in A \times C \Rightarrow x = \langle y, z \rangle$, където $y \in A, z \in C$,

$x \notin B \times C \Rightarrow x = \langle y, z \rangle$, където $y \notin B, z \in C$.

Следователно $y \in (A \setminus B), z \in C \Rightarrow x = \langle y, z \rangle \in (A \setminus B) \times C$

Обяснение: Ако $z \in C$ да докажем, че $\overline{B} \times C \leftrightarrow \overline{B \times C}$

тогава и само тогава, когато $y \in \overline{B}$

1. Нека $x = \langle y, z \rangle \in \overline{B} \times C \Rightarrow y \in \overline{B}$

2. Ако $x \in \overline{B \times C} \Rightarrow$ то $x \notin B \times C$. Но $x = \langle y, z \rangle$ и $z \in C \Rightarrow y \notin B$

Следователно:

$$\overline{B \times C} = \overline{B} \times C \text{ и } (A \setminus B) \times C = (A \times C) \setminus (B \times C)$$

Задача 4. Да се докаже, за A, B, C, D непразни множества твърдството:

$$A \subseteq B, C \subseteq D \leftrightarrow A \times C \subseteq B \times D$$

Доказателство:

I. Нека $A \subseteq B, C \subseteq D$ (1)

$$x \in A \times C \Rightarrow \langle y, z \rangle \in A \times C \Rightarrow y \in A, z \in C$$

От (1) $\Rightarrow y \in B, z \in D$

Следователно $\langle y, z \rangle \in B \times D$

II. Нека $A \times C \subseteq B \times D$

Ако $y \in A$, ще докажем че $y \in B$

Ако $A \times C \subseteq B \times D$, то съществува $\langle y, z \rangle \in A \times C$, където $y \in A, z \in C$ и $\langle y, z \rangle \in B \times D$, където $y \in B, z \in D$

Приемаме, че $x \in A$. От горното следва, че $x \in B$. Аналогично от $x \in C \Rightarrow x \in D$.

Задача 5. Докажете, че декартовото произведение не е асоциативно чрез пример.

$$A = \{\text{ябълки, круши}\}$$

$$B = \{\text{червени, зелени}\}$$

$$C = \{\text{пакети, колети}\}$$

$$A \times (B \times C) \neq (A \times B) \times C$$

$$A \times (B \times C) = \{\langle \text{ябълки, } \langle \text{червени, пакети} \rangle \rangle, \\ \langle \text{ябълки, } \langle \text{зелени, пакети} \rangle \rangle, \\ \text{-----} \\ \langle \text{круши, } \langle \text{зелени, колети} \rangle \rangle\}.$$

Описва четири начина за опаковане на два вида плодове.

$$(A \times B) \times C = \{\langle \langle \text{ябълки, червени} \rangle, \text{пакети} \rangle, \\ \langle \langle \text{ябълки, зелени} \rangle, \text{пакети} \rangle, \\ \text{-----} \\ \langle \langle \text{круши, зелени} \rangle, \text{колети} \rangle\}.$$

Описва два начина за опаковане на четири вида плодове.

Задача 6. За отношенията R_1 и R_2 да се докаже, че

$$(R_1 \cup R_2)^{-1} = R_1^{-1} \cup R_2^{-1}$$

Доказателство:

Нека $\langle x, y \rangle \in (R_1 \cup R_2)^{-1} \leftrightarrow \langle y, x \rangle \in (R_1 \cup R_2)$

От $\langle y, x \rangle \in R_1 \cup R_2 \leftrightarrow \langle y, x \rangle \in R_1 ; \langle y, x \rangle \in R_2$

1. $\langle y, x \rangle \in R_1 \leftrightarrow \langle x, y \rangle \in R_1^{-1} \leftrightarrow \langle x, y \rangle \in R_1^{-1} \cup R_2^{-1}$

2. $\langle y, x \rangle \in R_2 \leftrightarrow \langle x, y \rangle \in R_2^{-1} \leftrightarrow \langle x, y \rangle \in R_2^{-1} \cup R_1^{-1}$

От 1 и 2 $\Rightarrow \langle x, y \rangle \in R_1^{-1} \cup R_2^{-1}$

Задача 7. За всяко бинарно отношение R да се докаже равенството $\overline{R^{-1}} = (R)^{-1}$

Доказателство: Допълнение на бинарното отношение R между множествата A и B се нарича множеството:

$$\overline{R} = (A \times B) \cap \overline{R} = (A \times B) \setminus R$$

Нека $\langle x, y \rangle \in \overline{R^{-1}} \leftrightarrow \langle x, y \rangle \in (B \times A) \cap \overline{R^{-1}} \leftrightarrow$

$x \in B, y \in A, \langle y, x \rangle \notin R \leftrightarrow \langle y, x \rangle \in (A \times B) \cap \overline{R} \leftrightarrow$

$\langle x, y \rangle \in (\overline{R})^{-1}$

Задача 8. Да се докаже, че за бинарните отношения R_1 и R_2 е в сила $(R_1 \cdot R_2)^{-1} = R_1^{-1} \cdot R_2^{-1}$.

Доказателство:

$\langle a, b \rangle \in (R_1 \cdot R_2)^{-1} \leftrightarrow \langle b, a \rangle \in R_1 \cdot R_2 \leftrightarrow$ съществува с такова, че $\langle b, c \rangle \in R_1, \langle c, a \rangle \in R_2 \leftrightarrow$ съществува с такова, че $\langle a, c \rangle \in R_2^{-1}, \langle c, b \rangle \in R_1^{-1} \leftrightarrow \langle a, b \rangle \in R_2^{-1} \cdot R_1^{-1}$.

Задача 9. Да се докаже, че композицията (произведението) на симетричните отношения R_1 и R_2 е симетрично тогава и само тогава, когато $R_1 \cdot R_2 = R_2 \cdot R_1$.

Доказателство:

Необходимо условие :

Приема се ,че $(R_1.R_2)$ е симетрично, да се докаже, че $R_1.R_2=R_2.R_1$.

Щом $R_1.R_2$ е симетрично, следва $(R_1.R_2)=(R_1.R_2)^{-1}$. От задача 8 $(R_1.R_2)^{-1} = R_2^{-1}.R_1^{-1} \Rightarrow R_1.R_2 = R_2.R_1$

Достатъчно условие:

Нека $R_1.R_2 = R_2.R_1$

Следователно $(R_1.R_2)^{-1} = (R_2.R_1)^{-1} \Rightarrow$ следва от равенството $(R_2.R_1)^{-1}=R_1^{-1}.R_2^{-1} \Rightarrow$ от задача 8 $R_1^{-1}.R_2^{-1} = R_1.R_2 \Rightarrow$ от симетричност на R_1 и на R_2 .

Упражнение 3.

Изчисления с низове. Операции с низове.

Регулярни множества и изрази.

Теоретичен увод

Определения

Подредени множества - множества, чиито елементи са подредени в дума **STRING**. Съществуват понятията първи, преден, заден, последен и т.н. елемент. Подреденото множество изразява повече информация от неподреденото.

Азбука - произволно множество от символи:

$\Sigma = \{a, b, \dots, z\}$ символите се наричат букви на азбуката.

Обозначения:

$t, u, v, w, \dots$ - Дума (низ) над Σ - всяка крайна редица от букви;

$\wedge$ - Празна дума - дума, която не съдържа букви;

$A, B, C, \dots$ Множество от думи;

Σ^* - Множеството от всички думи над Σ включително и $\wedge$;

Σ^+ - Множеството от всички думи над Σ без $\wedge$.

Операции с низове.

Съединяване (конкатенация) на низове.

Нека $U \subseteq \Sigma^*$, $V \subseteq \Sigma^*$

$UV = \{x \mid x = uv, u \in U, v \in V\}$, т.е. всяка дума от UV се получава чрез присъединяване на дума от U с дума от V .

Пример:

$U = \{a, ab, aab\}$,

$V = \{b, bb\} \rightarrow UV = \{ab, abb, aabb, abb, abbb, aabbb\}$,

$VU = \{ba, bab, baab, \dots\}$.

Обединяване на низове е еквивалентно на обединяване на множества от низове:

Нека $U \subseteq \Sigma^*$, $V \subseteq \Sigma^*$

$U \cup V = \{x \mid x = u \cup v, u \in U, v \in V\}$.

Пример:

$$U = \{a, ab, aab\},$$

$$V = \{b, bb\} \rightarrow U \cup V = \{a, ab, aab, b, bb\}.$$

Затворена обвивка (звезда) на дадено множество S се нарича множеството, състоящо се от $\wedge$ и всички думи, образувани от съединяване на краен брой думи от S .

$$S^* = S^0 \cup S^1 \cup S^2 \cup \dots \cup S^{i-1} \cup S^i,$$

където $S^0 = \{\wedge\}$, $S^i = S^{i-1}S$, ($i > 0$).

Регулярни множества

Клас множества от думи над Σ , който се получава кратен брой прилагания на следните правила:

- всяко крайно множество на думи над Σ (включително празното множество) е регулярно множество;

- ако U и V са регулярни множества над Σ , то тяхното обединение и конкатенация също са регулярни множества;

- ако S е регулярно множество над Σ , затворената му обвивка S^* също е регулярно множество.

Пример: Множеството от думи над $\Sigma = \{a, b\}$, съдържащо aa или bb е регулярно.

$$T = \{aa, aab, abb, abba, \dots\}.$$

Нерегулярно е $T = \{a^n b^n \mid n > 0\}$, защото не е крайно.

Регулярни изрази - служат за представяне на регулярни множества и алгебрични манипулации с тях. Класът на регулярните изрази над Σ се дефинира рекурсивно по следните правила:

1. $\wedge$ и празното множество $\emptyset$ са регулярни изрази над Σ .
2. Всяка буква $\delta \in \Sigma$ е регулярен израз над Σ .
3. Ако R_1 и R_2 са регулярни изрази над Σ , то $(R_1 + R_2)$, $(R_1 \cdot R_2)$ и R_1^* също са регулярни изрази.

Всеки регулярен израз R описва регулярно множество $\bar{R}$ от думи над Σ , т.е. $\bar{R} \subseteq \Sigma^*$:

1. $R = \wedge$ описва $\tilde{R} = \{\wedge\}$ - множеството състоящо се от празната дума.

$R = \emptyset$ описва $\tilde{R} = \emptyset$ - празното множество.

2. $R = \delta$, където $R \in \Sigma \rightarrow \tilde{R} = \{\delta\}$ - множество състоящо се от буква.

3. $R = R_1 + R_2 \rightarrow \tilde{R} = \tilde{R}_1 \cup \tilde{R}_2 = \{x \mid x \in \tilde{R}_1; x \in \tilde{R}_2\}$ - обединение на две множества.

$R = R_1 \cdot R_2 \rightarrow \tilde{R} = \tilde{R}_1 \cdot \tilde{R}_2 = \{xy \mid x \in \tilde{R}_1, y \in \tilde{R}_2\}$ - конкатенация на две множества

$R = (R_1)^* \rightarrow \tilde{R} = \tilde{R}_1^* = \{\wedge\} \cup \{x \mid x \text{ е получено чрез съединяване на краен брой думи от } \tilde{R}_1\}.$

Приоритет на операциите $()$, $*$, $.$, $+$.

Два регулярни изрази R_1 и R_2 над Σ са еквивалентни

(означава се с $R_1 = R_2$) тогава и само тогава, когато $\tilde{R}_1 = \tilde{R}_2$.
Например $b(ab)^* = (ba)^*b$ са еквивалентни.

Правила за еквивалентност на РИ:

$$1. R+S = S+R, R+\wedge = \wedge+R, R+R=R, (R+S)+T = R+(S+T).$$

$$2. R = \wedge R = R, R\wedge = \wedge R = \wedge, (RS)T = R(ST).$$

$$3. R(S+T) = RS+RT, (S+T)R = SR+TR.$$

$$4. R^* = R^*R^* = (R^*)^* = (\wedge+R)^*, \wedge^* = \wedge^* = \wedge.$$

$$5. R^* = \wedge + R + R^2 + \dots + R^k + R_{k+1}R^* \quad (k \geq 0),$$

в частност $R^* = \wedge + RR^*$.

$$6. (R+S)^* = (R^*+S^*)^* = (R^*S^*)^* = (R^*S)^*R^* = R^*(SR^*)^*,$$

в общия случай $(R+S)^* \neq R^*+S^*$.

$$7. R^*R = RR^*, R(SR)^* = (RS)^*R.$$

$$8. (R^*S)^* = \wedge + (R+S)^*S, (RS^*)^* = \wedge + R(R+S)^*.$$

$$9. \text{Правило на Ардън. Нека } \wedge \notin S.$$

Тогава:

$$R = SR+T \leftrightarrow R = S^*T;$$

$$R = RS+T \leftrightarrow R = TS^*.$$

Примерни задачи

*Задача1. Зададена е азбуката Σ , определена от множеството думи $\{a,b\}$. Съставете множество от думи с дължина 4.

Дадено $\Sigma = \{a,b\}$. Да се определи Σ^* и Σ^+ при $|\Sigma^*| = 4$.

Решение:

$$\Sigma^* = \{aaaa, aaab, aaba, abaa, aabb, \dots\}.$$

*Задача2. Зададена е азбуката Σ , определена от множеството символи $\{0,1\}$.

а) Да се състави множество от символи с дължина 4 и всеки получен низ да се превърне в число от шестнадесетичната система.

б) Да се състави конкатенация от двойки низове.

Решение:

а)	$0000 \rightarrow 0$	б)	00
	$0001 \rightarrow 1$		01
	$0010 \rightarrow 2$		02
			
	$1111 \rightarrow F$		FF

*Задача3. Зададено е множество от низове $A = \{a, ab, aab, aba, abb\}$ и $B = \{b, ba, bba, bab, baa\}$.

Да се съставят посредством конкатенация множествата $C = A.B$ и $D = B.A$.

Решение:

$$C = A.B = \{ab, abb, aabb, abab, \dots, abbbbaa\}.$$

$$D = B.A = \{ba, bab, baab, baba, \dots, baaabb\}.$$

Задача4. Дадено е множеството $S = \{ab, ba\}$, да се образува S^ .

Решение:

Затворената обвивка на множеството S се образува чрез обединеното множество от множествата низове с дължина

$0, 1, 2, \dots$, съответно $S^0, S^1, S^2, \dots$, т.е.:

$$S^* = S^0 \cup S^1 \cup S^2 \cup \dots,$$

където

$$S^0 = \{\wedge\}$$

$$S^1 = \{ab, ba\}$$

$$S^2 = \{abab, abba, baab, baba\} \text{ и т.н.}$$

Изобщо $S^i = S^{i-1} \cdot S$ за $i \geq 0$, т.е. множеството от думи с дължина i се получава чрез конкатенация на множеството думи с дължина $i-1$ и множеството S .

***Задача5.** Да се определи какви множества от думи над азбуката $\Sigma = \{a, b\}$ описват регулярните изрази:

- а) ab^* - започват с a и следват само b -та;
- б) $b^*ab^*ab^*$ - съдържат 2 a -та;
- в) $(a+b)^*$ - всички думи $\Sigma = \{a, b\}$ т.е. Σ^* ;
- г) $(a+b)^*(aa+bb)(a+b)^*$ - всички думи, съдържащи 2 последователни a или 2 последователни b ;
- д) $[bb+aa+(ba+ab)(bb+aa)^*(ab+ba)]$ - четен брой a и четен брой b ;
- е) $(a+baa)^*$ - всяко b е следвано от минимум 2 a ;
- ж) $(ba)^*b$ - алтернативно сменящи се a и b ;

***Задача6.** Да се докаже равенството $R(SR)^* = (RS)^*R$

Решение:

а) Нека $w \in R(SR)^*$ е произволен елемент, принадлежащ на лявата част на равенството. Следователно може да се образува низ с дължина $n \geq 0$ от букви (низове):

$$r_i \in R \text{ и } s_i \in S,$$

$$w = r_0(s_0r_1)(s_1r_2)\dots(s_{n-1}r_n).$$

Поради асоциативността на операцията конкатенация, низът може да се представи във вида:

$$w = (r_0 \cdot s_1)(r_1 \cdot s_2) \dots (r_{n-1} \cdot s_n)r_n,$$

откъдето следва, че

$$w \in (RS)^*R, \text{ т.е. } R(SR)^* \subseteq (RS)^*R.$$

б) Нека $w \in (RS)^*R$

$$w = (r_0 s_1)(r_1 s_2) \dots (r_{n-1} s_n) r_n, \text{ което се представя във}$$

вида:

$$w = r_0 (s_1 r_1) (s_2 r_2) \dots (s_{n-1} r_n), \text{ т.е.}$$

$$w \in R(SR)^* \Rightarrow$$

$$(RS)^*R \subseteq R(SR)^*.$$

*Задача 7. Да се докаже правилото на Ардън.

Ако $\wedge \notin S$, то $R = SR + T \Leftrightarrow R = S^*T$.

Доказателство:

Приема се, че $R = SR + T$

а) да се докаже $R = S^*T$

$$\text{От } R = SR + T \Rightarrow R = S(SR + T) + T \Rightarrow S^2R + ST + T \Rightarrow S^2(SR + T) + ST + T \Rightarrow S^3R + S^2T + ST + T.$$

$$1. R = S^{k+1}R + (S^kT + S^{k-1}T + \dots + ST + T) \text{ за всяко } k \geq 0.$$

Нека $x \in R$ и $|x| = l$.

Щом равенството 1. е вярно за всяко $k \geq 0$, то е вярно за $k=l$. Следователно:

$$R = S^{l+1}R + (S^lT + S^{l-1}T + \dots + ST + T), \text{ по определение}$$

$$S^*T = (S^lT + S^{l-1}T + \dots + ST + T), \text{ за низ с дължина } l.$$

Щом $\wedge \notin S$ всяка дума от $S^{l+1}R$ трябва да има поне $l+1$ символа следователно $x \notin S^{l+1}R$, но

$$x \in R \Rightarrow x \in S^lT + S^{l-1}T + \dots + ST + T \Rightarrow x \in S^*T, \text{ т.е.}$$

$$R \subseteq S^*T.$$

б) да се докаже $S^*T \subseteq R$

Нека $x \in S^*T$. Тогава съществува $i \geq 0$, такова, че $x \in S^iT$.

От равенството 1. :

$$R = S^{i+1}R + (S^iT + S^{i-1}T + \dots + ST + T) \Rightarrow R \subseteq S^iT \Rightarrow x \in R$$

*Задача8. Да се опрости с помощта на формулите:

$$\begin{aligned}
 R &= (b+aa*b)+(b+aa*b)(a+ba*b)*(a+ba*b) \\
 \Rightarrow &(b+aa*b)(\wedge+(a+ba*b)(a+ba*b)^*) \\
 \Rightarrow &(\wedge+aa^*)b(a+ba*b)^* \\
 \Rightarrow &a*b(a+ba*b)^*.
 \end{aligned}$$

*Задача9. Да се опростят с помощта на формулите:

$$\wedge+b*(abb)^*(b*(abb)^*)^*.$$

*Задача10. Да се докаже, че

$$(a+b(ba)^*(bb+a))^*b = a*b((ba)^*(bb+a)a^*b)^*.$$

*Задача11. Да се докаже равенството:

$$\begin{aligned}
 &(ba)^*b+(ba)^*.(bb+a)(b(ba)^*(bb+a)+a)^*b(ba)^*b = \\
 &= (ba+(bb+a)a^*b)^*b.
 \end{aligned}$$

Упражнение 4.

Формални граматик и езици.

Теоретичен увод

Определения

Формален език L : множество от низове с крайна дължина на азбуката Σ , т.е. $L \subseteq \Sigma^*$

Описание на езика L . Езикът L може да бъде описан по два начина:

1.Чрез изреждане: използва се при малка мощност на множеството думи.

2.Чрез правила за образуване: когато мощността на множеството думи е голяма или при безкраен език. Безкрайният език се описва с краен брой правила за образуване.

Формалното средство за описание на езика L се нарича формална граматика.

Формална граматика G .

$G = \{\Sigma, N, P, S\}$, където:

Σ - множеството от букви(азбуката) на, описан с граматиката G ;

N - крайно, непразно множество от символи, наречени нетерминални(променливи):

$N \cap \Sigma = \emptyset$ - двете множества не се пресичат;

$N \cup \Sigma = U$ - нарича се речник на езика L ;

S - начална променлива $S \in N$, т.е. S е нетерминален символ (начален символ);

P - крайно множество от правила ,определящи как се пораждат всичките думи на езика L . Всяко правило има вида $\alpha \rightarrow \beta$, където $\alpha \in (N \cup \Sigma)^+$, а $\beta \in (N \cup \Sigma)^*$).

Пораждане(генериране) на дума $w \in \Sigma^*$ от граматиката G имаме тогава , когато съществува крайна редица от думи $S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \dots \Rightarrow w_n$, $n \geq 1$, за която w_{i+1} , се получава от

$w_i (0 \leq i \leq n)$ като заместим подниз α в w_i с β , използвайки някое правило от P .

Класификация на граматиките.

Таблица 4.1. Класификация на граматики

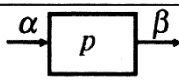
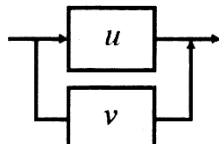
Тип	Название	Ограничения към правилата
0	без ограничения	$\alpha \rightarrow \beta$ Няма (според определението на формална граматика)
1	контекстно зависима	$ \alpha \leq \beta $ (дължините на лява и дясна част)
2	контекстно свободна	$\alpha \in N, \beta \in (N \cup \Sigma)^+$
3	регулярна	$A \rightarrow \sigma B; A \rightarrow \sigma$, където $A, B \in N, \sigma \in \Sigma$ или $\sigma = \Delta$

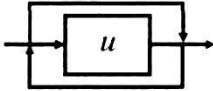
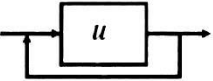
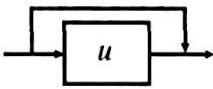
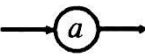
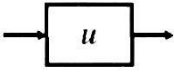
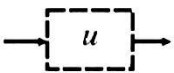
БНФ форма на описание. Синтактични графи

Определение. БНФ е форма за описание на продукциите представлява граматика за описание на граматиките.

Ако $u, v \in \Sigma \cup N$, тогава имаме следния запис на правилата и синтактичните графи.

Таблица 4.2. БНФ и синтактични графи

Обяснение	Запис	Синтактичен граф
$\alpha \rightarrow \beta$	$\alpha ::= \beta (\alpha \Rightarrow \beta)$	
конкатенация	uv	
алтернатива	$u \setminus v$	

итерация $u^n \mid n \geq 0$	$u^*, (u)^*, \{u\}$	
итерация $u_n \mid n > 0$	$u^+, (u)^+, uu^*$	
$u_n \mid n = 0, 1$	$[u^+]$	
терминален символ $a \in \Sigma$	$'a'$	
нетерминален символ	u	
помощни скоби	(u)	

Примерни задачи

Задача 1. Да се определят Σ, N, P, S на език L , разпознаващ следните четири изречения:

- а) студентите учат;
- б) студентите почиват;
- в) студентките учат;
- г) студентките почиват.

Решение: Синтактичните правила, описващи езика L са следните:

- изречение $\rightarrow$ подлог казуемо
- подлог $\rightarrow$ студентите
- подлог $\rightarrow$ студентките

сказуемо $\rightarrow$ учат
сказуемо $\rightarrow$ почиват

Правят се следните полагания:

изречение	S	студентките	b
подлог	A	учат	c
сказуемо	B	почиват	d
студентите	a		

Граматиката $G = \{\Sigma, N, P, S\}$ се дефинира по следния начин:

$$\Sigma = \{a, b, c, d\}, N = \{S, A, B\}, S = S, P = \{p_i \mid i = 1, 5\}$$

$$\begin{array}{ll} p_1: S \rightarrow AB & p_4: B \rightarrow c \\ p_2: A \rightarrow a & p_5: B \rightarrow d \\ p_3: A \rightarrow b & \end{array}$$

Задача 2. Дадена е азбуката $\Sigma = \{a, b, c\}$. Да се дефинира граматиката $G = \{\Sigma, N, S, P\}$, която поражда езиците:

$$\begin{array}{l} \text{а) } L = \{a^i b^i c^i \mid i \geq 1\} \\ \text{б) } L = \{a^i b^k c^k \mid i, k \geq 1\} \\ \text{в) } L = \{a^i b^k c^e \mid i, k, l \geq 1\} \end{array}$$

Да се определи типа на граматиката.

Решение:

$$\begin{array}{ll} \text{а) } p_1: F \rightarrow aFBC & p_4: bB \rightarrow bb \\ p_2: F \rightarrow abC & p_5: bC \rightarrow bc \\ p_3: CB \rightarrow BC & p_6: cC \rightarrow cc \end{array}$$

Обяснение: Прилага се правило p_1 $i-1$ пъти, където $i \geq 1$ и се получава $a^{i-1} F(BC)^{i-1}$. Използва се веднъж правило p_2 и се получава $a^i bC(BC)^{i-1}$. Правилото p_3 позволява да се пренаредят B и C и да се получи $a^i bB^{i-1}C^i$, като се приложи $i-1$ пъти. Прилагането на правило p_4 $i-1$ пъти преобразува думата в $a^i b^i C^i$. Посредством правилото p_5 се получава $a^i b^i cC^{i-1}$.

Последното правило се прилага $i-1$ път и се получава $a^i b^i c^i$.
Граматиката G на езика L се дефинира чрез:

$$\Sigma = \{a, b, c\}, N = \{F, B, C\}, P = \{p_i \mid i = 1, 6\}, S = F.$$

Тя е от тип G_1 (контекстно зависима), понеже не удовлетворява условията на G_3 и G_2 .

б) Правилата на езика са следните:

$$\begin{array}{ll} p_1: F \rightarrow aF & p_3: E \rightarrow bEc \\ p_2: F \rightarrow aE & p_4: E \rightarrow bc \end{array}$$

Обяснение:

Правило p_1 се прилага $i-1$ път и се получава $a^{i-1}F$. Правило p_2 , приложено веднъж образува $a^i E$. Посредством правило p_3 (лява и дясна рекурсия на b и c) приложено $k-1$ пъти се образува думата $a^i b^{k-1} E c^{k-1}$. Последно се прилага граматичното правило p_4 и се получава $a^i b^k c^k$.

Граматиката $G = \{\Sigma, N, P, S\}$ се определя от :

$$\Sigma = \{a, b, c\}, N = \{F, E\}, S = F, P = \{p_i \mid i = 1, 4\}$$

Типът на граматиката е G_2 (контекстно свободна), тъй като не удовлетворява условията на G_3 .

в) Правилата на езика са следните:

$$\begin{array}{ll} p_1: S \rightarrow aS & p_4: A \rightarrow bB \\ p_2: S \rightarrow aA & p_5: B \rightarrow cB \\ p_3: A \rightarrow bA & p_6: B \rightarrow c \end{array}$$

Обяснение: Правилото p_1 се прилага $i-1$ пъти и се получава $a^{i-1}S$. По p_2 се получава $a^i A$. Прилага се $k-1$ път правилото p_3 и се получава $a^i b^{k-1} A$. По p_4 се завършва поредицата b -та и се въвежда променливата B в низа. Правило p_5 се прилага $l-1$ пъти и с p_6 се завършва търсеният низ $a^i b^k c^l$.

Граматиката $G = \{\Sigma, N, P, S\}$ се дефинира чрез:

$\Sigma = \{a, b, c\}$, $N = \{S, A, B\}$, $P = \{p_i \mid i=1,6\}$, $S = S$.
Типът на граматиката е G_3 (регулярна).

Задача 3. Дадена е граматиката $G = \{\Sigma, N, P, S\}$

$\Sigma = \{A_i \mid i \geq 1, -, \cap, \cup, \setminus, \times, \subset, \subseteq, =, (,)\}$, $N = \{M\}$, $S = M$,
 $P = \{p_i \mid i = 1, 8\}$

$$p_1: M \rightarrow A_i$$

$$p_6: M \rightarrow (M \subset M)$$

$$p_2: M \rightarrow \bar{M}$$

$$p_7: M \rightarrow M \times M$$

$$p_3: M \rightarrow (M \cup M)$$

$$p_8: M \rightarrow M = M$$

$$p_4: M \rightarrow (M \setminus M)$$

$$p_9: M \rightarrow (M \cap M)$$

$$p_5: M \rightarrow (M \subseteq M)$$

Кой език поражда граматиката G ?

Решение: Езикът за описание на операции с множества:

A_i -символи за обозначаване на множествата;

M -множество.

Правилата определят допустимите операции между множества:

p_2 - допълнение (отрицание) на множеството;

p_3 - обединение;

p_4 - разлика;

p_5 - подмножество;

p_6 - собствено множество;

p_7 - декартово произведение;

p_8 - равенство на множества;

p_9 - сечение.

Задача 4. Съставете БНФ запис и синтактични графи за граматиката от задача 3.

Решение:

Въвеждат се нетерминални символи:

- израз_в_скоби - обхваща правилата p_3, p_4, p_5, p_6, p_9 ;

- оператор - за оператори $\cup, \setminus, \subseteq, \subset, \cap$.

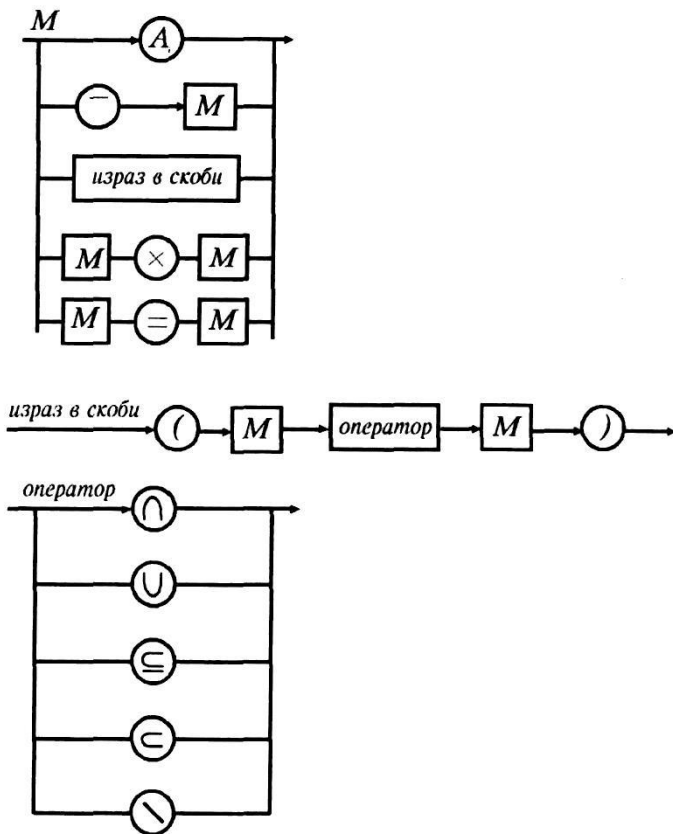
Правилата на граматиката в БНФ имат вида:

$M \Rightarrow 'A_i' \mid '-'M \mid \text{израз_в_скоби} \mid M'X'M \mid M'=M$

- израз в скоби $\Rightarrow '(M \text{ оператор } M)'$;

- оператор $\Rightarrow '\setminus' \mid '\cup' \mid 'X' \mid '\subseteq' \mid '\cap'$.

Синтактични графи:



Задача 5. Дадена е БНФ форма на езика L . Определете елементите на езика и постройте синтактични графи.

$$R \Rightarrow M[E]$$

$$M \Rightarrow [V]D$$

$$E \Rightarrow 'E'[V]I$$

$$V \Rightarrow '+' | '-'$$

$$D \Rightarrow D_0 | D_i$$

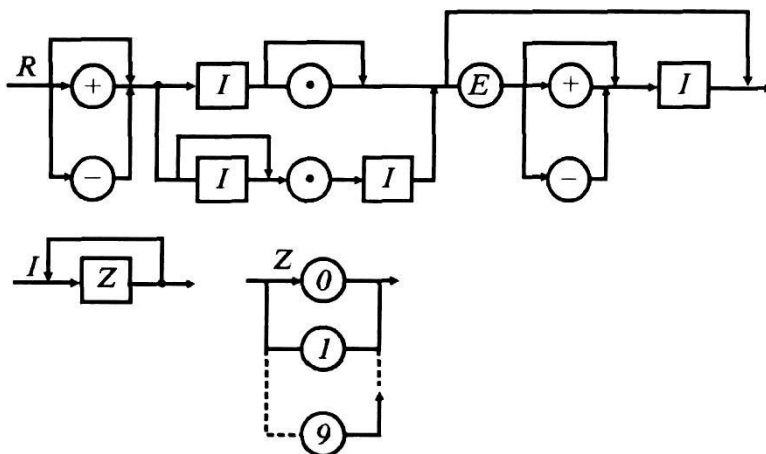
$$D_0 \Rightarrow I['.']$$

$$D_i \Rightarrow [I]'.I$$

$$I \Rightarrow Z^+$$

$$Z \Rightarrow '0' | '1' | '2' | \dots | '9'$$

Решение:



R - число с плаваща запетая обикновена точност;

M - мантиса;

E - експонента;

V - знак;

D - нецяло число;

D_0 - форма на представяне на нецяло число;

D_i - форма на представяне на нецяло число;

I - цяло число;

Z - цифра.

Упражнение 5.

Краини автомати - анализ.

Теоретичен увод

Определения

Краен ориентиран граф: множество, състоящо се от $\{V, D\}$, където $V = \{V_i \mid I \geq 1\}$ е множество от върхове V_i (крайно множество). $D = \{ \langle V_i, V_j \rangle \mid i, j \geq 1 \}$ е крайно множество от наредени двойки (дъги) между върхове V_i и V_j .

Път от V_i до V_k е редицата $V_i, V_2, \dots, V_k$ ако всяка наредена двойка $\langle V_i, V_{i+1} \rangle$, където $I \leq i \leq k$ е насочена дъга на КОГ.

Краен автомат над азбуката $\Sigma = \{a_1, a_2, \dots, a_n\}$ се нарича КОГ, с етикет a_i ($I \leq i \leq n$). Един връх (начален) е отбелязан със знак '-' и множество (евентуално празно) от върхове са означени с '+' (крайни върхове). Началният и краен връх могат да съвпадат. Върховете се наричат състояния на крайния автомат.

Диаграма на преходите: КОГ, на който всяка насочена дъга е отбелязана с дума. $W \in \Sigma^*$ (евентуално на Λ), наречена още етикет. Има един връх отбелязан със знак '-' и (евентуално празно) множество от върхове, белязани с знак '+' (крайни върхове). Даден връх може да бъде начален и краен.

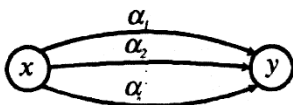
W път от V_i до V_j : Ако $w \in \Sigma^*$, W път се нарича такъв краен път от връх V_i до връх V_j , конкатенацията от етикетите на насочените дъги по който представлява думата W . (ако има празни думи по пътя, те се пренебрегват) Казваме, че една дума $w \in \Sigma^*$ се разпознава от диаграма на преходите, ако съществува W път от начален до краен връх. Празната дума се разпознава от диаграма на преходите, ако съществува Λ път между начален и краен връх, или съществува връх, който едновременно е и начален и краен.

Обобщена диаграма на преходите: Диаграма на преходите, чиито насочени дъги могат да бъдат регулярни изрази.

Алгоритъм за построяване на регулярен израз за дадена диаграма на преходите, така че $\bar{R} = \bar{T}$

Към диаграмата на преходите се прибавя нов връх x и Λ дъги, водещи от x към началния връх на T и нов връх y с Λ дъги, водещи от всеки краен връх към y . Елиминират се стъпка по стъпка всички върхове на T , докато останат само новите върхове x и y . По време на този процес дъгите и техните етикети се променят.

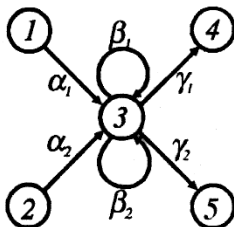
Процесът завършва когато се получи обобщена диаграма на преходите само с два върха x и y , от вида:



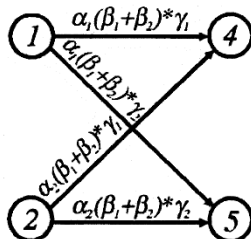
Фиг. 5.1. Обобщена диаграма на преходите

Тогава $R = \alpha_1 + \alpha_2 + \dots + \alpha_n$

Нека е необходимо елиминиране на върха 3 в следната обобщена диаграма на преходите:



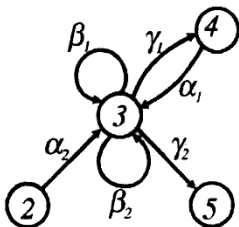
Фиг. 5.2. Примерна диаграма



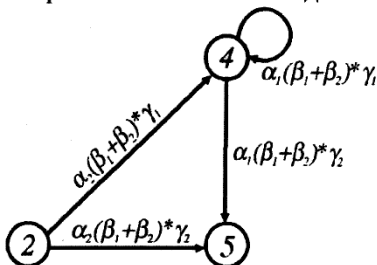
Фиг. 5.3. Елиминиран връх 3

$\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2$ - регулярни изрази.

В частния случай, когато върховете 1 и 4 съвпадат:



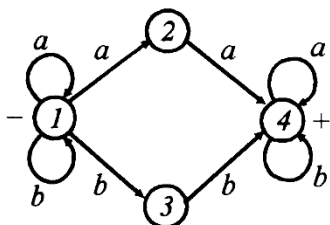
Фиг. 5.4. Примерна диаграма



Фиг. 5.5. Елиминиран връх 3

Примерни задачи

Задача 1: Езикът L се представя чрез следния КОГ:

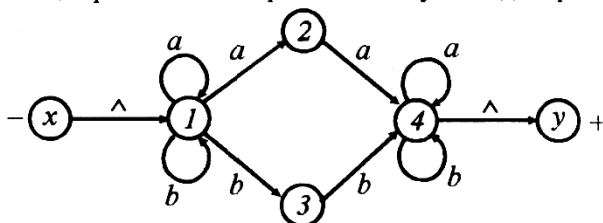


Да се представи L чрез регулярен израз.

Решение:

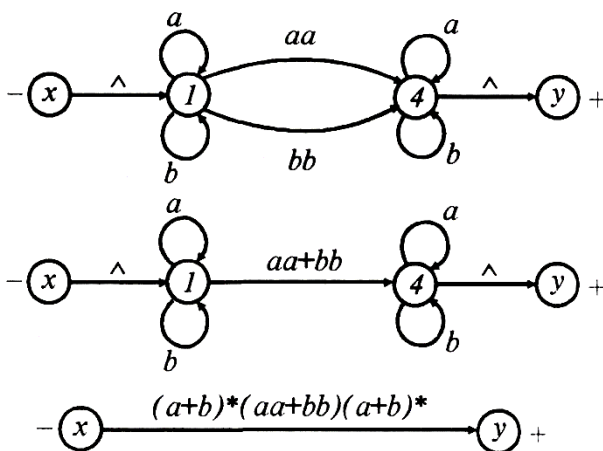
Фиг. 5.6. Краен ориентиран граф

а) Прибавят се върховете x и y към диаграмата



Фиг. 5.7. Добавяне на върхове

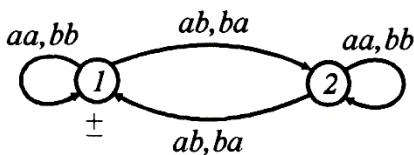
Последователно се елиминират върховете 2,3,1,4:



$$R = (a+b)^*(aa+bb)(a+b)^*$$

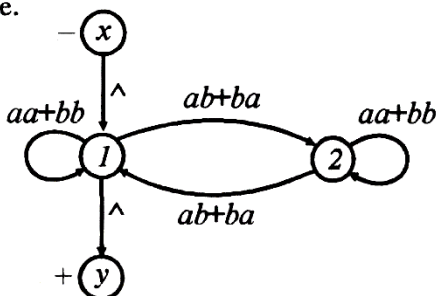
Фиг. 5.8. Получаване на обобщена диаграма

Задача 2. Да се определи регулярният израз R , зададен със следната диаграма на преходите:



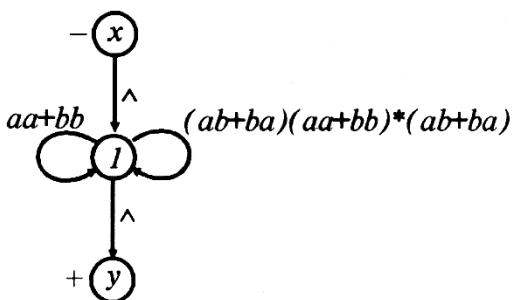
Фиг. 5.9. Диаграма на преходите

Решение: Прибавят се нови върхове x и y към диаграмата на преходите.



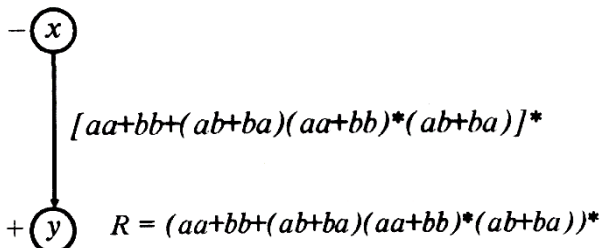
Фиг. 5.10. Добавяне на върхове

Елиминира се връх 2 и се получава:



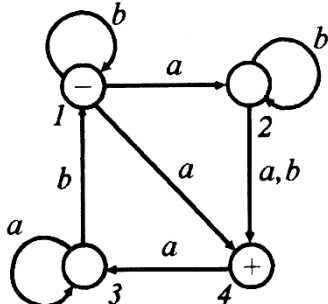
Фиг. 5.11. Елиминирание на връх

Елиминира се връх 1:



Фиг. 5.12. Обобщена диаграма

Задача 3. Езикът L се разпознава от следния краен ориентиран граф:



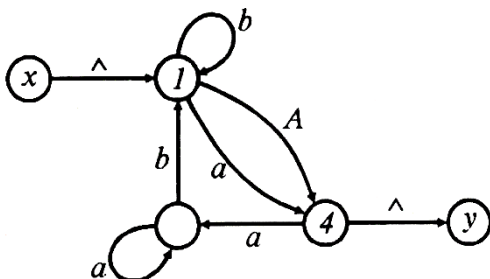
Да се представи L чрез :

- регулярен израз
- регулярна граматика

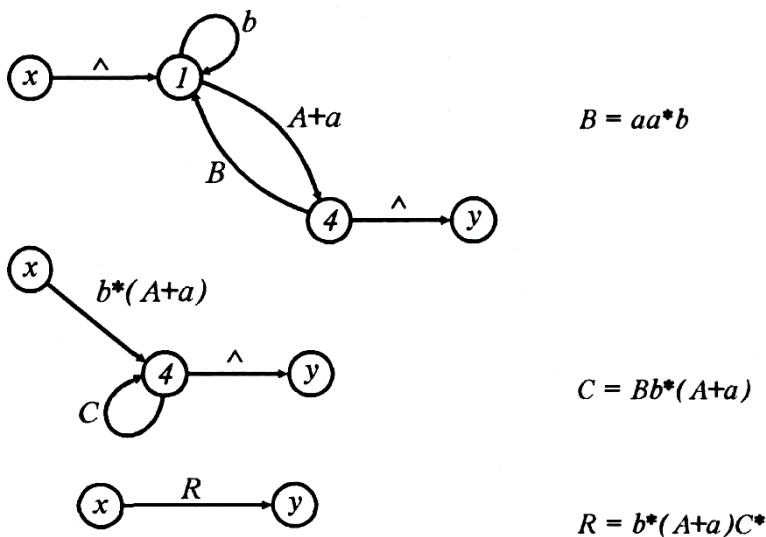
Фиг. 5.13. Краен ориентиран граф

Решение:

а) Преобразува се КОГ, като последователно се елиминират върхове 2,3,1,4 .



$$A = ab^*(a+b)$$



Фиг. 5.14. Елиминирание на върхове

Полученият регулярен израз, съответстващ на КОГ е:

$$R = b^*(ab^*(a+b) + a)(aa^*bb^*(ab^*(a+b) + a))^*$$

б) Съставя се регулярна граматика:

$$p_1: S \rightarrow bS \quad p_6: C \rightarrow aA$$

$$p_2: S \rightarrow aB \quad p_7: C \rightarrow bS$$

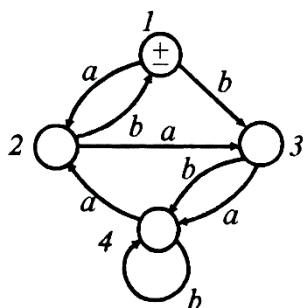
$$p_3: B \rightarrow bB \quad p_8: C \rightarrow \Lambda$$

$$p_4: B \rightarrow aC \quad p_9: A \rightarrow aA$$

$$p_5: B \rightarrow \Lambda \quad p_{10}: A \rightarrow bS$$

$$G = \{ \Sigma, N, P, S \} \quad \Sigma = \{ a, b \} \quad N = \{ S, A, B, C \} \quad P = \{ p_i \mid i=1, 10 \}$$

Задача 4. Езикът L се разпознава от следния КОГ:

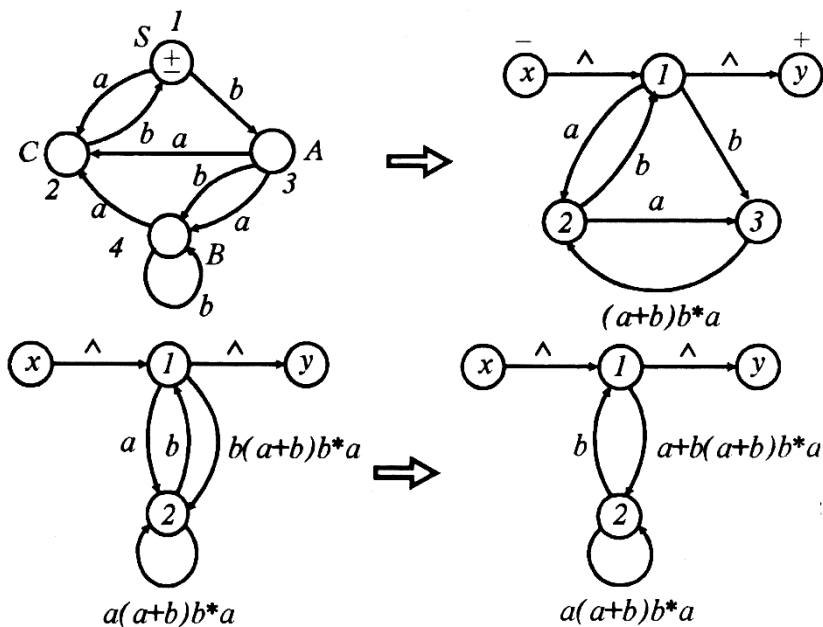


Представете L чрез
а) регулярен израз;
б) формална граматика.

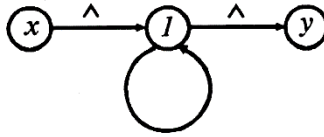
Фиг. 5.15. Краен ориентиран граф

Решение:

а) Преобразува се КОГ чрез последователно елиминиране на върховете 4, 3, 2, 1.



Фиг. 5.16. Елиминиране на върхове



$$(a+b(a+b)b^*a)(a(a+b)b^*a)^*b$$

Фиг. 5.17. Обобщена диаграма на преходите

б) Регулярният израз се преобразува до получаване на граматиката:

$$R = ((a + b(a + b)b^*a)(a(a + b)b^*a)^*b)^*$$

По правилото Ардън:

$$R = \wedge + (a + b(a + b)b^*a)(a(a + b)b^*a)^*bR$$

Полага се нетерминал: $C = (a(a + b)b^*a)bR$

$$R = \wedge + (a + b(a + b)b^*a)C$$

Прилага се формулата за регулярни изрази

$$(P + Q)R = PR + QR$$

$$R = \wedge + aC + b(a + b)b^*aC$$

Полага се: $A = (a + b)b^*aC = (a + b)B$, където $B = b^*aC$

$$R = \wedge + aC + bA$$

Образува се граматиката:

$$p_1: R \rightarrow aC$$

$$p_2: R \rightarrow bA$$

$$p_3: R \rightarrow \wedge$$

$$p_4: A \rightarrow aB$$

$$p_5: A \rightarrow bB$$

$$p_6: B \rightarrow bB$$

$$p_7: B \rightarrow aC$$

$$p_8: C \rightarrow aA$$

$$p_9: C \rightarrow bR$$

$$G = \sum, N, P, S$$

$$\sum = \{a, b\}$$

$$N = \{R, A, B, C\}$$

$$S = R$$

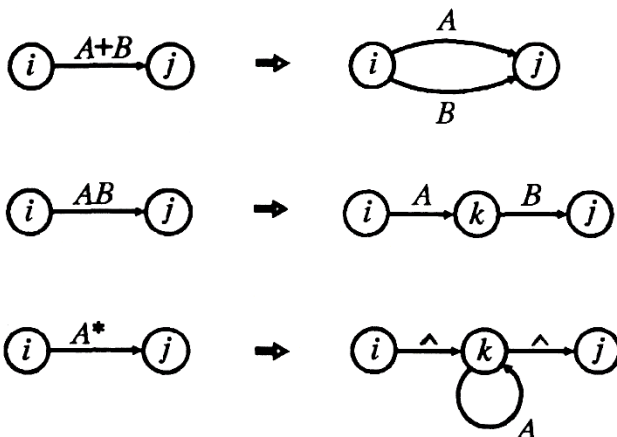
$$P = \{P_i | i = 1, 9\}$$

Упражнение 6. Крайни автомати. Синтез.

Теоретичен увод

Преобразуване на регулярен израз в краен автомат

Въвеждат се върховете x - начален и y - краен, които се свързват с насочена дъга от x към y с обозначение зададения регулярен израз. Последователно се разклонява израза като се въвеждат нови върхове и дъги, докато всяка дъга остане обозначена само с буква от азбуката на езика или празен низ. използват се следните правила за разчленяване на изрази:



Фиг. 6.1. Преобразуване на регулярен израз в краен автомат

Получената диаграма на преходите се преобразува в краен автомат:

Съставя се таблица на преходите, съставена от три стълба (за азбука от 2 символа a , b). Елементите на таблицата са подмножества от върхове на диаграмата. M - множество от I-ви стълб с начална стойност, началния връх на диаграмата.

M_a - множество от върхове, до които има дъги с обозначения a от кой да е връх на множеството в I-ва колона на същия ред.

M_b - множество от върхове, до които има дъги с

обозначения в от кой да е връх на множеството в I-ва колона на същия ред.

Новополучените множества се добавят на следващия ред в I-ва колона, ако не се срещат преди това там.

Процесът завършва при изчерпване на множествата.

Крайният автомат се съставя по получената таблица. Той има толкова върхове, колкото са редовете на таблицата. Дъгите се образуват чрез наредените двойки $\langle M, Ma \rangle$ и $\langle M, Mb \rangle$. Начален е върхът, съответстващ на множеството в I-ви ред I-ва колона. Крайни са всички върхове, които включват в множеството си краен връх на диаграмата.

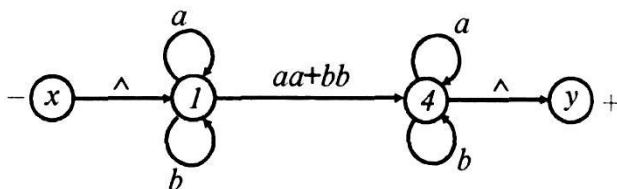
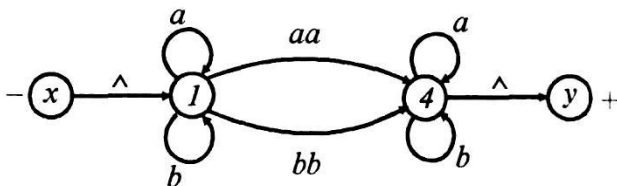
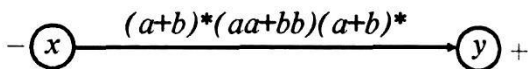
Примерни задачи

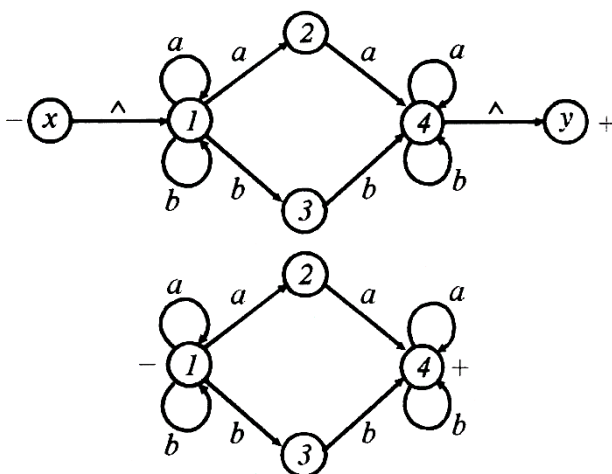
Задача 1. Даден е регулярен израз:

$$R = (a + b)^*(aa + bb)(a + b)^*$$

Да се състави краен автомат по регулярен израз.

Решение:





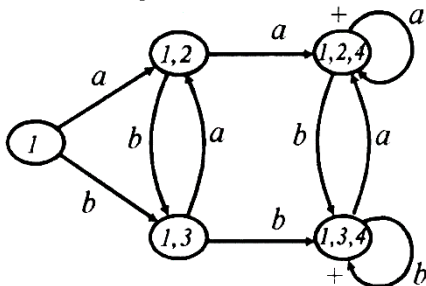
Фиг. 6.2. Краен ориентиран граф

Съставя се таблица на преходите за получената диаграма:

Таблица 6.1. Таблица на преходите

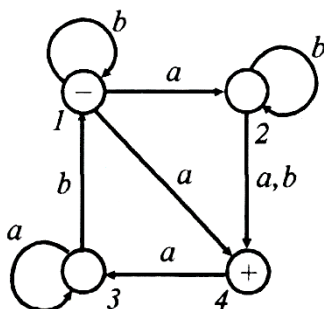
M	M_a	M_b
$\{1\}$	$\{1,2\}$	$\{1,3\}$
$\{1,2\}$	$\{1,2,4\}$	$\{1,3\}$
$\{1,3\}$	$\{1,2\}$	$\{1,3,4\}$
$\{1,2,4\}$	$\{1,2,4\}$	$\{1,3,4\}$
$\{1,3,4\}$	$\{1,2,4\}$	$\{1,3,4\}$

Съответстващият краен автомат е:



Фиг. 6.3. Краен автомат

Задача 2. Езикът L е зададен чрез КОГ. Да се преобразува в краен автомат.



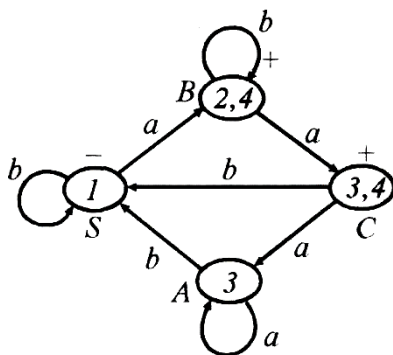
Фиг. 6.4. Краен ориентиран граф

Съставя се таблица на преходите за КОГ:

Таблица 6.2. Таблица на преходите

M	M_a	M_b
$\{1\}$	$\{2,4\}$	$\{1\}$
$\{2,4\}$	$\{3,4\}$	$\{2,4\}$
$\{3,4\}$	$\{3\}$	$\{1\}$
$\{3\}$	$\{3\}$	$\{1\}$

По таблицата на преходите се построява КА.



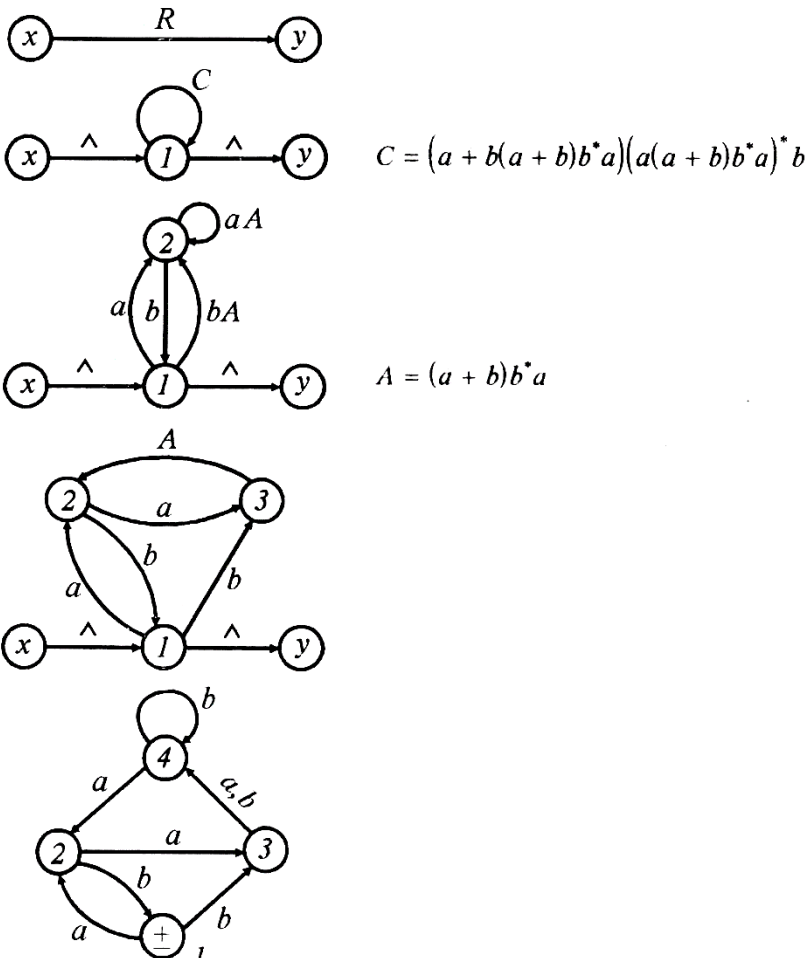
Фиг. 6.5. Краен автомат

Задача 3. Езикът L е зададен чрез регулярен израз:

$$R = \left((a + b(a + b)b^*a) (a(a + b)b^*a)^* b \right)^*$$

Да се представи L чрез краен автомат.

Решение:

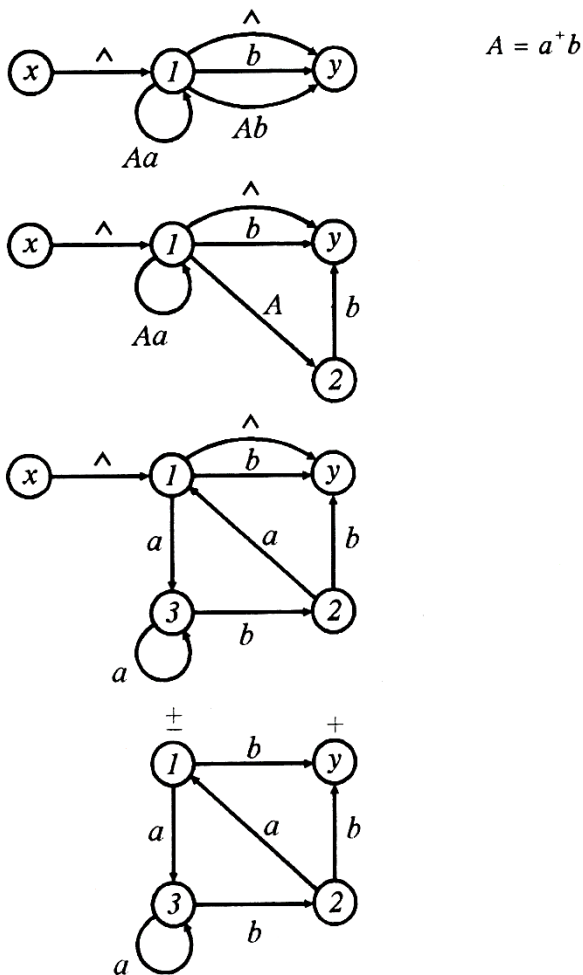


Фиг. 6.6. Краен автомат за зад.3

Задача 4. Езикът L е зададен чрез регулярен израз. Да се представи L чрез краен автомат

$$R = (a^+ba)^*(\wedge + b + a^+bb).$$

Решение:



Фиг. 6.7. Краен автомат за зад. 4

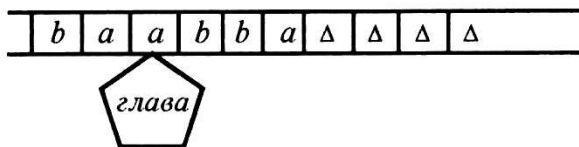
Упражнение 7. Машины на Тюринг.

Дефиниция

Машината на Тюринг над азбуката Σ се състои от 3 части:

Лента, която е разделена на клетки. Тя има най-лява клетка, но е безкайна отдясно. Всяка клетка на лентата съдържа точно един лентов символ. Лентовите букви включват буквите от дадената азбука Σ , буквите от помощната азбука V и специален символ за празна клетка Δ . Всички символи са различни.

Глава (четящо-записваща), която във всеки момент прочита съдържанието на една клетка от лентата. Това устройство може да се придвижва като при всеки ход записва на мястото на прочетения символ (евентуално) нов символ, след което се придвижва с една клетка наляво или надясно.



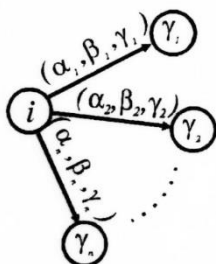
Програма, която е КОГ (върховете се наричат състояния). Винаги има начално състояние, означено със START (или -) и (евентуално празно) подмножество от крайни състояния, означени с HALT (или +). Всяка насочена дъга има вида:



В началото дадената дума w над Σ се намира в лявата част на лентата, а останалите клетки съдържат символа Δ , т.е. са празни, а главата е върху символа в най-лявата клетка. Изпълнението протича според инструкциите на програмата, като

се започва от състоянието START. Ако в крайна сметка се достигне до състоянието HALT, машината спира работа и се

казва, че машината на Тюринг разпознава думата w . Ако се достигне до състояние i от вида, при което главата е върху символа, различен от $\alpha_1, \alpha_2, \dots, \alpha_n$, при което МТ не може да продължи работа, тогава машината отхвърля думата w . Другият случай при който машината спира работа и казваме, че отхвърля думата w е, когато главата е върху най-левия символ и инструкцията е придвижване наляво.



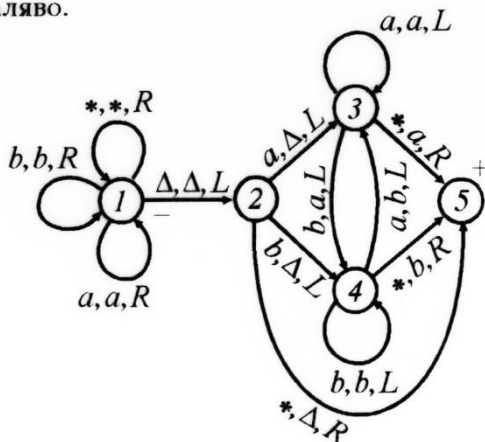
Примерни задачи

Задача 1. Да се построи машина на Тюринг над $\Sigma = \{a, b, *\}$, която да извършва действието конкатенация:

Вход $w_1 * w_2 \Delta$ - изход $w_1 w_2 \Delta$

Да се покаже действието при входна лента $ab * ba \Delta$ - изход $abba \Delta$.

Решение: Алгоритъм: Движим се надясно, докато достигнем Δ , връщаме се наляво, докато достигнем $*$, като изместваме w_2 наляво.



Фиг. 7.1. КОГ в машина за Тюринг



Фиг. 7.2. Работа на МТ над входна лента

Тук низа от вида $ab*ba\Delta$ представлява текущото съдържание на лентата, а стрелката сочи символа, върху който е главата. Стрелката $abba\Delta$ от вида $\xrightarrow{b,a,L}$ означава коя инструкция от програмата се изпълнява, а кръгчето с номер, като например (1), означава текущото състояние.

Задача 2. Да се състави машина на Тюринг, над азбуката $\Sigma=\{0,1\}$, която да измества всички нули вляво, а всички единици - вдясно. Да се покаже действието на машината при вход 0101Δ .

Решение 1:

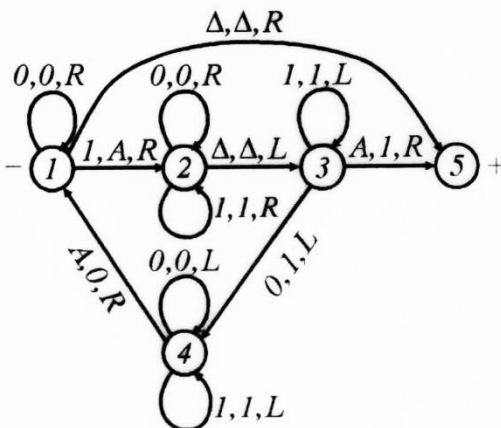
Алгоритъм:

1) Движим се надясно до първата единица, маркираме с помощен символ A . При поява на Δ - край;

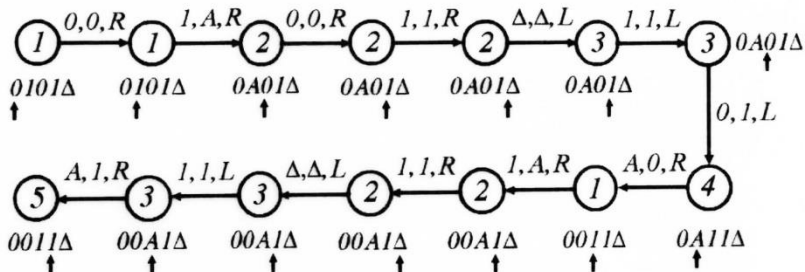
2) Движим се надясно до Δ ;

3) Движим се наляво до нула. При поява на A , заменяме с 1 и край. Нулата се заменя с 1 ;

4) Движим се наляво до A , заменяме с 0 и повтаряме цикъла - 1.



Фиг. 7.3. КОГ в МТ за решение 1



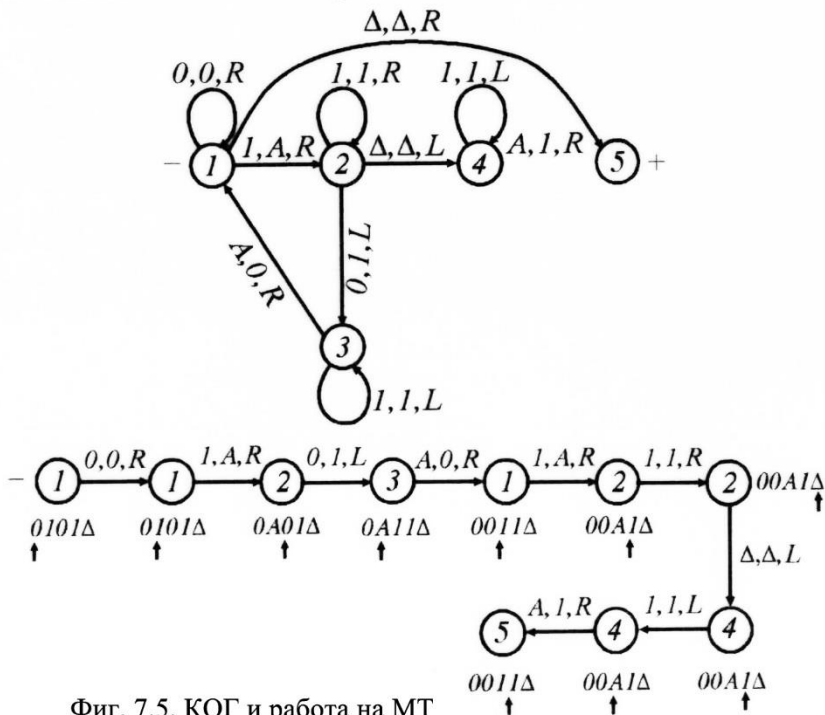
Фиг. 7.4. Работа на МТ

Решение 2:

Алгоритъм:

1) Движим се надясно до първата 1. Заменя се с А и надясно. При поява на Δ , наляво до А и замяна с 1.

2) Движим се до първата нула, заменяме с 1 и наляво до А. Заменяме А с 0. Повтаря се 1.



Фиг. 7.5. КОГ и работа на МТ

Задача 3. Да се построи стеков автомат (СА) над азбуката $\Sigma = \{a, b\}$, разпознаващ всяка дума w на езика $L = \{a^n b^n \mid n \geq 0\}$

и отхвърлящ всички останали. $reject(M) = \Sigma^* - accept(M)$

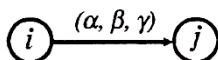
$loop(M) = \emptyset$

Операции над стека S при $\Sigma = \{a, b\}$

1) $push(x)$ - въвеждане на символ в стека

2) $pull(x)$ - извежда се символ от стека

Търси се КОГ

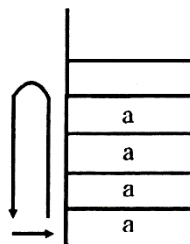
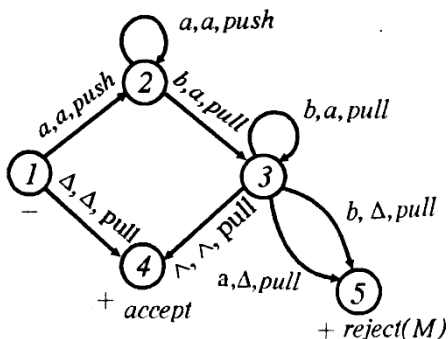
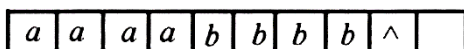


α - символ върху лентата ,

β - символ в или от стека ,

$\gamma = push | pull$; $push(S) = \beta$; $\beta = pull(S)$

Решение:



Фиг. 7.6. Стеков автомат за зад. 3

Упражнение 8.
Мрежи на Петри. Графи на мрежи на Петри.
Анализ и синтез.

Теоретичен увод

Определения

Структура на мрежа на Петри - C

Описва се с множеството $C = \{P, T, I, O\}$, където

C -структура

P -множество от места

T -множество на преходите

I -множество от входни функции

O -множество от изходни функции

I и O описват отношението между преходи и места.

Входна функция на прехода. За произволен преход t_j , функцията $I(t_j)$ присвоява на прехода t_j подмножества от P , наречени входни места на прехода. Съдържанието на множеството места, принадлежащи на прехода t_j се определя от функцията $I(t_j)$.

Изходна функция на прехода. За произволен преход t_j , функцията $O(t_j)$ присвоява на прехода t_j подмножества от P , наречени изходни места на прехода. Съдържанието на множеството места, принадлежащи на прехода t_j се определя от функцията $O(t_j)$.

Граф на мрежа на Петри G : Насочен двумножествен граф, състоящ се от множество върхове V и множество насочени дъги A , т.е. $G = \{V, A\}$,

където

$V = \{V_1, V_2, \dots, V_n\}$ - множество от върховете на G .

$A = \{ a_1, a_2, \dots, a_m \}$ - множество от насочени дъги на G .
 Всяко a_i може да се представи чрез наредена двойка:

$$a_i = \langle V_i, V_k \rangle,$$

където $V_i, V_k \in V$.

V се дели на две подмножества:

$V = \{P, T\}$ за което е изпълнено

$$V = P \cup T, P \cap T = \emptyset,$$

т.е. V е обединено множество на две множества:

P -множество места;

T -множество преходи.

Двете множества са различни (не се пресичат).

Ако $a_i = \langle V_i, V_k \rangle$, тогава $V_i \in P$ и $V_k \in T$ или $V_i \in T$ и $V_k \in P$.

Задачи

Задача 1. Дадена е структура на МП $C = \{P, T, I, O\}$,

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6\}, T = \{t_1, t_2, t_3, t_4, t_5\}$$

$$I(t_1) = \{p_1\}$$

$$O(t_1) = \{p_2, p_3\}$$

$$I(t_2) = \{p_3\}$$

$$O(t_2) = \{p_3, p_5, p_5\}$$

$$I(t_3) = \{p_2, p_3\}$$

$$O(t_3) = \{p_2, p_4\}$$

$$I(t_4) = \{p_4, p_5, p_5, p_5\}$$

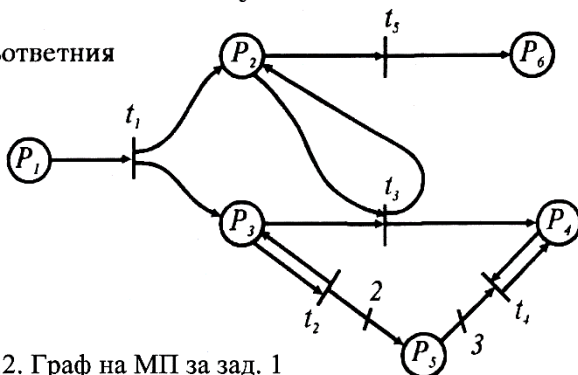
$$O(t_4) = \{p_4\}$$

$$I(t_5) = \{p_2\}$$

$$O(t_5) = \{p_6\}.$$

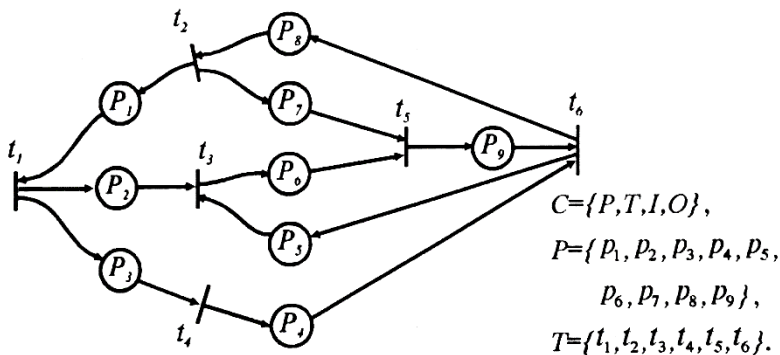
Съставете съответния
граф на МП.

Решение:



Фиг. 8.2. Граф на МП за зад. 1

Задача 2. Даден е граф на МП.



Фиг. 8.2. Граф на МП за зад. 2

Определете съответната структура.

Решение:

$I(t_1) = \{P_1\}$	$O(t_1) = \{P_2, P_3\}$
$I(t_2) = \{P_8\}$	$O(t_2) = \{P_1, P_7\}$
$I(t_3) = \{P_2, P_5\}$	$O(t_3) = \{P_6\}$
$I(t_4) = \{P_3\}$	$O(t_4) = \{P_4\}$
$I(t_5) = \{P_6, P_7\}$	$O(t_5) = \{P_9\}$
$I(t_6) = \{P_4, P_9\}$	$O(t_6) = \{P_5, P_8\}.$

Задача 3. Даден е фрагмент на Паскал.

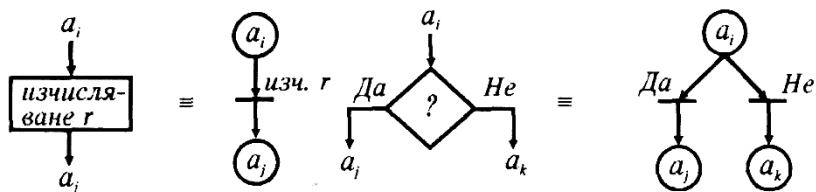
```

begin
    read(x1, x2); x3 = 1;
    while x1 > 0 do
        begin
            if odd(x1) then
                begin
                    x3 := x3 - x2; x1 := x1 - 1;
                end;
            x2 := x2 * x2; x1 := x1 - 2;
        end;
    write(x3);
end;

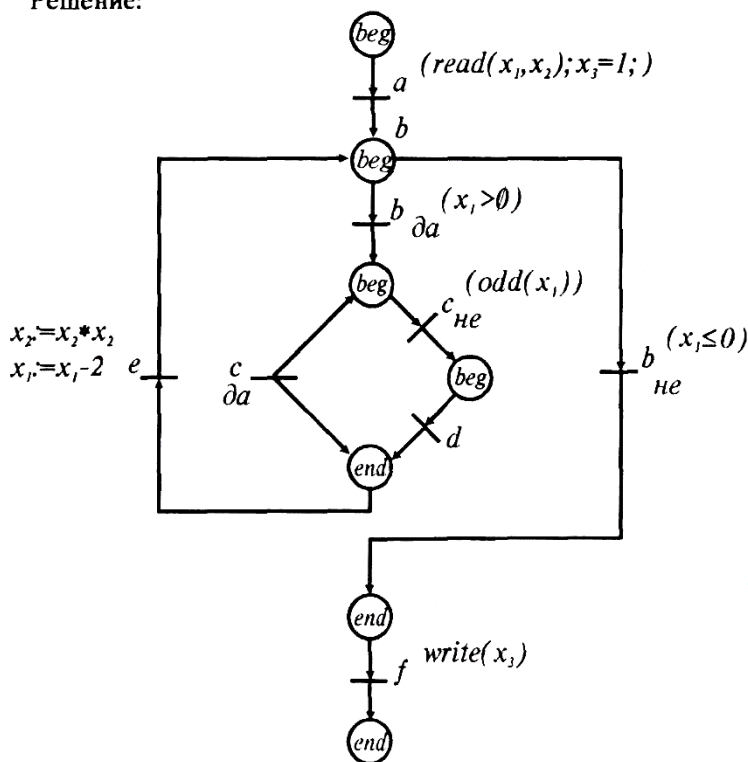
```

a
b
c
d
e
f

Използвайте съответствието между блокова схема и граф на МП и образувайте съответния граф.



Решение:



Фиг. 8.3. Граф на МП за зад.3

Упражнение 9.

Маркирани мрежи на Петри. Действие на мрежите.
Свойства на маркираните мрежи на Петри.

Теоретичен увод

Маркирани мрежи на Петри

Под маркирана мрежа на Петри разбираме присвояване на символи(марки) на местата в мрежата. Местата се разглеждат като памет(броячи), в които се запомня броя на присвоените марки. Броят на маркиите в мрежата се променя при работата (изчислението) с мрежата.

Маркирането се дефинира като n - мерен вектор.

$\mu = \{\mu_1, \mu_2, \dots, \mu_n\}$ n - мощност на множеството P ,
т.е. $n = |P|$.

Маркирана мрежа на Петри M . Множество от мрежа на Петри C и вектор μ .

$M = \{C, \mu\}$, където $C = \{P, T, I, O\}$, т.е.

$M = \{P, T, I, O, \mu\}$.

Марките в мрежата се представят чрез символ "." или с число N .

Работа на маркираната мрежа на Петри

Работата на мрежата се управлява от текущия брой и положение на маркиите в местата. Марките се местят от едно място на друго(съседно), преминавайки през преходите. Преходите пропускат маркиите през себе си когато са активни.

Активен преход: Ако във всяко входно място на прехода има поне толкова марки, колкото дъги има мястото до прехода, той се нарича активен. При наличие на мултидъги са необходими повече от една марка. Условието прехода t_j да бъде активен е:

$\mu(P_i) \geq \#(P_i, I(t_j))$, където $\#(P_i, I(t_j))$ е броят входни

дъги от мястото P_i до преход t_j .

При сработване на преход t_j се променя текущата маркировка $\mu(P_i)$ на мястото P_i в нова

$$\mu'(P_i) = \mu(P_i) - \#(P_i, I(t_j)) + (P_i, O(t_j)).$$

Маркировката на мястото P_i се намалява с толкова марки $\#(P_i, I(t_j))$, колкото са входните дъги от мястото P_i до активния преход t_j и се увеличава с толкова марки $\#(P_i, O(t_j))$, колкото са изходните дъги между активния преход t_j и мястото P_i .

Свойства на маркирани мрежи на Петри

Надеждност - ММП е надеждна ако броят на маркиите в местата никога не е по-голям от I

$$\sum_i \mu(P_i) \leq I \text{ за всички } P_i \mid i \leq 0, P_i \in P.$$

Ограниченост - ММП е ограничена ако общият брой маркировки в местата никога не е по-голям от константа k .

$$\sum_i \mu_j(P_i) \leq k \text{ за всички } P_i \mid i \geq 0, P_i \in P.$$

Консервативност - ММП е консервативна ако общият брой на марки в мрежата остава постоянен, равен на константа k

$$\sum_i \mu_j(P_i) = k \text{ за всички } P_i \mid i \geq 0, P_i \in P.$$

Живучест - ММП е живуча ако не изпада в състояние блокада.

Блокада - състояние на ММП, при което част от мрежата или цялата мрежа не работи, т.е. не сработват преходите.

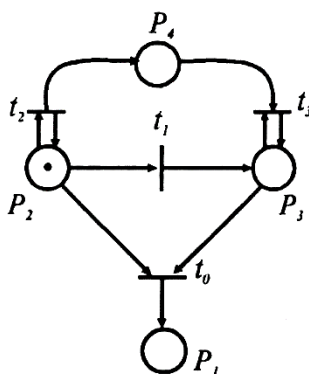
Достижимост - множество от достижимите текущи маркировки при работа на мрежата. Определя съществува ли.

$\mu' \in R(C, \mu)$, където R е множество от достижими маркировки от началното маркиране μ .

Покриваемост - Определя съществува ли маркировка μ'' спрямо маркировката $\mu' \in R(C, \mu)$, за която е изпълнено условието: $\mu'' \in R(C, \mu)$ и $\mu'' \geq \mu'$, т.е. марките да бъдат повече от тези в маркировка μ' .

Примерни задачи

Задача 1. Даден е граф на ММП. Анализирайте графа и определете кои преходи и колко пъти сработват. Какви свойства има мрежата.



t_0 - не сработва
 t_1 - само веднъж
 t_2 - n -пъти последователно преди сработване на t_1
 $t_3 = t_2$

Фиг. 9.1. Граф на ММП

Свойства:

1. Ненадеждна, защото не отговаря на условието

$$\sum_i \mu_j(P_i) \leq 1 \quad P_i \in P$$

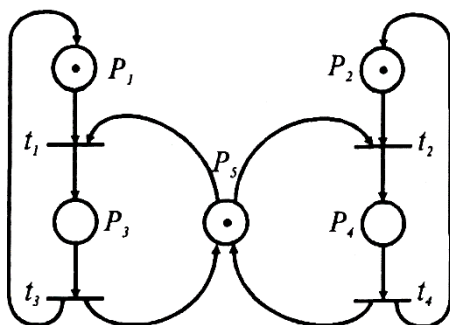
2. Неограничена, защото не отговаря на условието

$$\sum_i \mu_j(P_i) \leq k \quad P_i \in P$$

3. Неконсервативна

$$\sum_i \mu_j(P_i) \neq k \quad P_i \in P$$

Задача 2. Даден е граф на ММП, който не е консервативна. Да се преобразува в консервативна.



$$\sum_i \mu_j(P_i) = k = 3$$

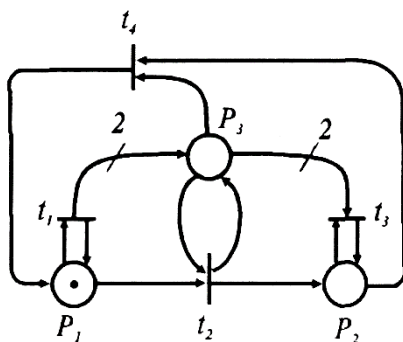
$$P_i \in P$$

Фиг. 9.2. Граф на ММП за зад.2

Решение:

Добавят се входни дъги от t_1 към P_3 , от t_2 към P_4 и изходни дъги от P_3 към t_3 и от P_4 към t_4 .

Задача 3. Даден е граф на ММП. Да се състави таблица на възможните преходи.



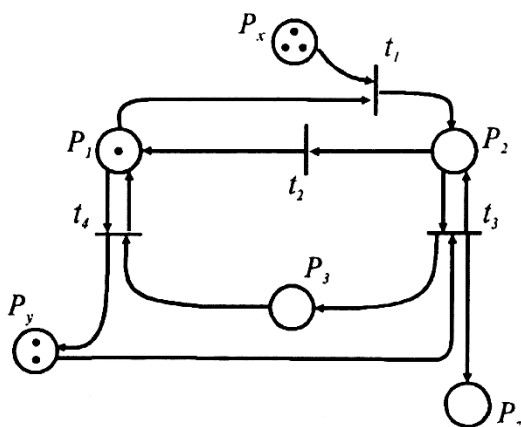
Фиг. 9.3. Граф на ММП за зад.3

Решение:

Таблица 9.1. Таблица на преходите

	P_1	P_2	P_3
и	1	0	0
t_1	1	0	2
t_2	0	1	2
t_3	0	1	0
t_4	блокада		

Задача 4. Даден е граф на ММП (t_2 с по-малък приоритет от t_3 и t_1 с по-малък приоритет от t_4).



Фиг. 9.4. Граф на ММП за зад.4

Колко марки ще има в P_i , когато настъпи блокада?
Коя функция изчислява мрежата?

Упътване: Съставете таблица на възможните преходи до настъпване на блокада. Анализирайте таблицата за определяне на функцията.

Отговор: $P_i = P_x * P_y = 3 * 2 = 6$

Упражнение 10. Мрежови езици.

Теоретичен увод

Обозначените маркирани мрежи на Петри разпознават:

- всички видове регулярни езици;
- контекстно свободни езици от вида:

$$L(C) = \{X^n W Y^n \mid n \geq 0, W \in \Sigma^*\}.$$

Пример:

$$L(C) = \{a^n b^n \mid n \geq 0\}$$

- контекстно зависими езици:

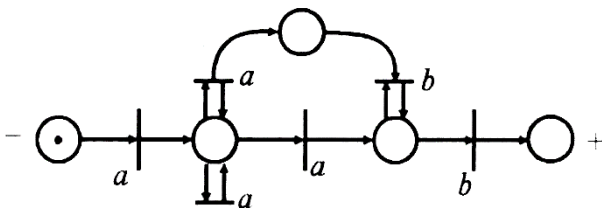
$$L(C) = \{X^n W_1 Y^n W_2 Z^n \mid n \geq 0, W_1, W_2 \in \Sigma^*\}.$$

Пример:

$$L(C) = \{a^n b^n c^n \mid n \geq 0\}.$$

Примерни задачи:

Задача 1. Дадена е обозначена ММП. Кой език $L(C)$ се поражда от мрежата?



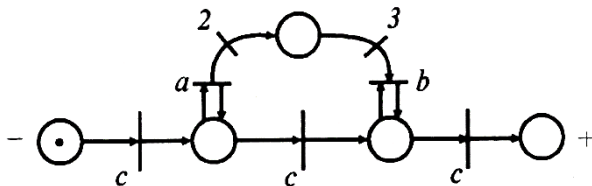
Фиг. 10.1. Граф на ОММП

Упътване: Да се анализира работата на мрежата при преминаване на марката от началното място до крайното, предполагайки сработване на всички възможни преходи.

Решение:

$$L(C) = \{a^m b^n \mid m > n > 1\}.$$

Задача 2. Дадена е обозначена ММП. Кой език $L(C)$ се поражда от мрежата?

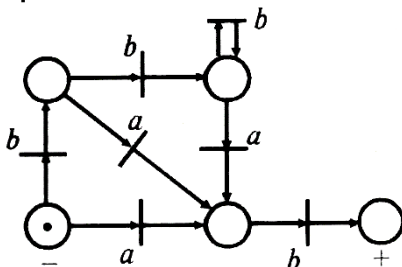


Фиг. 10.2 Граф на ОММП за зад.2

Решение:

$$L(C) = \{ca^{3n}cb^{2n}c \mid n \geq 0\}$$

Задача 3. Дадена е обозначена ММП. Кой език $L(C)$ се поражда от мрежата?



Фиг. 10.3 Граф на ОММП за зад.3

Решение:

$$L(C) = ab + bab + b^+ab = b^+ab$$

Задача 4. Даден е език

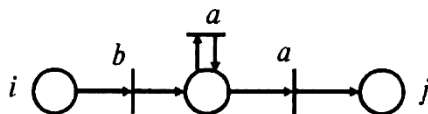
$$L(C) = \{(ba^k)^n c(ab^l)^n c \mid k, l \geq 1, n \geq 0\}$$

Да се състави ОММП, която да поражда езика. Да се състави граматика за $L(C)$.

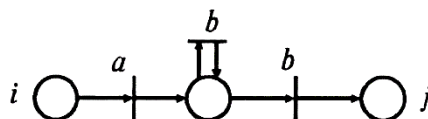
Решение:

а) Съставяне на ОММП:

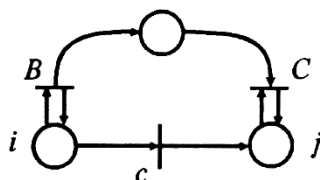
$B = ba^t$ се представя с подмрежата:



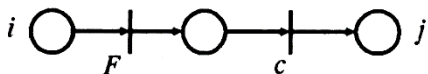
$C = ab^l$ се представя с подмрежата:



$F = B^n c C^n$:

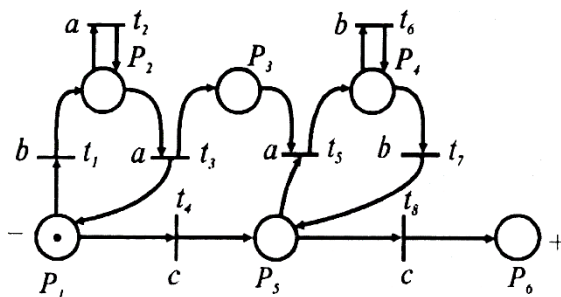


$S = Fc$



Фиг. 10.4. Подмрежи на ОММП

Елементите се свързват в обща мрежа:



Фиг. 10.5. Цялостна ОММП

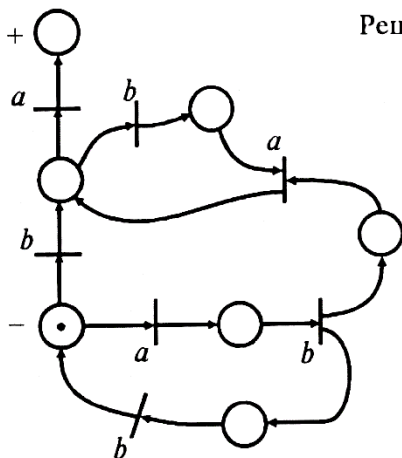
б) Съставяне на граматика:
Правилата са следните:

$$\begin{array}{ll} S \rightarrow Fc & D \rightarrow a \\ F \rightarrow BFC & C \rightarrow aE \\ F \rightarrow c & E \rightarrow bE \\ B \rightarrow bD & E \rightarrow b \\ D \rightarrow aD \end{array}$$

Задача 5. Дадена е ОММП. Да се определи $L(C)$ и се състави граматика за нея.

Решение:

$$L(C) = \{(abb)^n b(ba)^n a \mid n \geq 0\}.$$



Фиг. 10.6. Граф на ОММП за зад.5

$$\begin{array}{ll} p1: S \rightarrow Aa; & \\ p2: A \rightarrow BAC; & \text{или} \\ p3: A \rightarrow b; & \\ p4: B \rightarrow aD; & \\ p5: D \rightarrow bb; & \\ p6: C \rightarrow ba; & \end{array}$$

$$\begin{array}{ll} p1: S \rightarrow Aa; & \\ p2: A \rightarrow BAC; & \\ p3: A \rightarrow b; & \\ p4: B \rightarrow abb; & \\ p5: C \rightarrow ba. & \end{array}$$

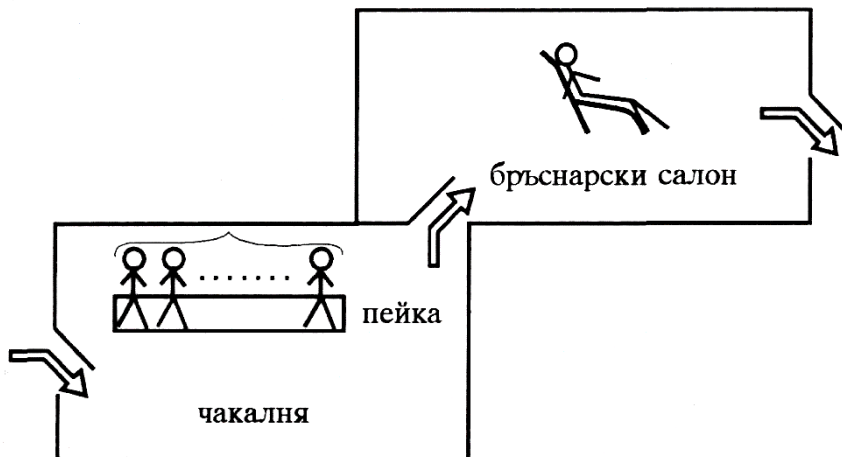
Упражнение 11.

Моделиране на паралелни процеси.

Задача 1. "Система за обслужване".

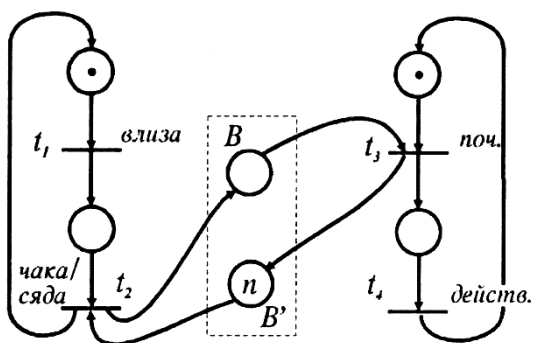
Постановка на задачата.

Да се състави ОММП, която да описва действието на системата.



Фиг. 11.1 Моделиране на система за обслужване

Решение:



Обозначения:

B - чакащи в
чакалнята;

B' - свободни
места на
пейката;

t_1 - влиза;

t_2 - сяда и чака;

t_3 - почива;

t_4 - действия по
обслужването;

Фиг. 11.2. Граф на ОММП

Задача 2. "Телефонна централа" (ТЦ)

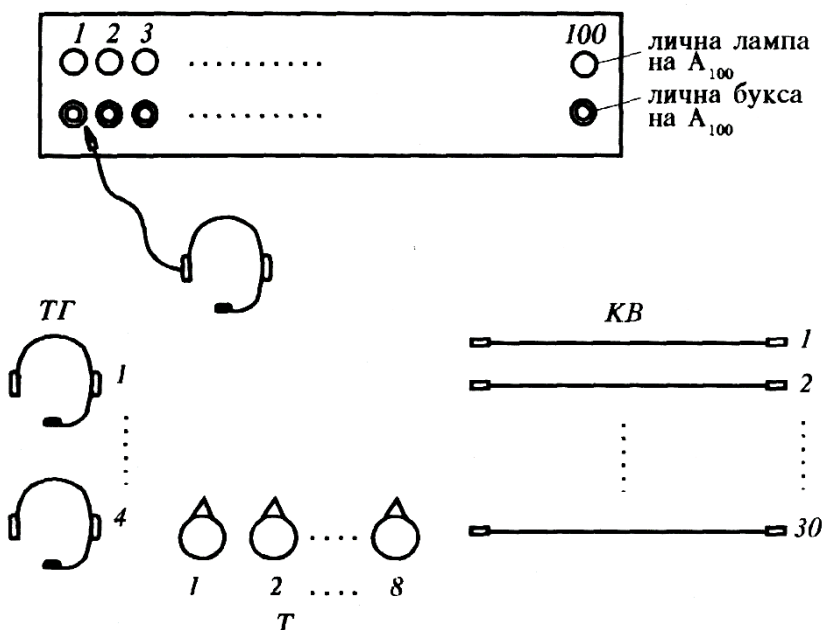
Дадено: Телефонна централа за 100 абоната (А) с ръчно обслужване. ТЦ разполага с 30 кабела за връзка (КВ), като едновременно са възможни 15 разговора. ТЦ разполага с 4 телефонни гарнитури (ТГ), т.е. телефонистките могат да говорят едновременно с 4-ма абонати. В ТЦ работят общо 8 телефонистки (Т).

Поведение на абонатите: Абонатът може да отсъства. Абонатът може да е у дома си но да няма желание сам да започне разговор. Абонатът е у дома си и желае да се обади на някого.

Описание на ТЦ: Буксите на абонатите са поместени в общо табло и при вдигане на слушалката (заявка) от абоната светва личната му сигнална лампа. КВ и ТГ са ресурси необходими за обслужване на абонатите. Т са ресурси играещи централна роля при обслужване на цялостната дейност в ТЦ.

Поведение на Т: Свободните телефонистки наблюдават таблото. При заявка за връзка една от тях (ако има свободна) се снабдява с ТГ (ако има свободна), влиза във връзка с абоната и получава номера на партньора. Ако партньора е зает, връща ТГ и се освобождава. В противен случай се снабдява с 2 КВ (ако има свободни) и се опитва да се свърже с партньора. Ако той не се обади, връща КВ и ТГ и се освобождава. Ако той се обади осъществява връзката с КВ, връща ТГ и се освобождава. При заявка за разпадане на връзката (единият абонат поставя слушалката) някоя свободна телефонистка (ако има такава) освобождава КВ и се освобождава.

Постановка в ТЦ:



Фиг. 11.3. Моделиране на телефонна централа

Да се моделира поведението на абоната i и на телефонистките чрез ОММП.

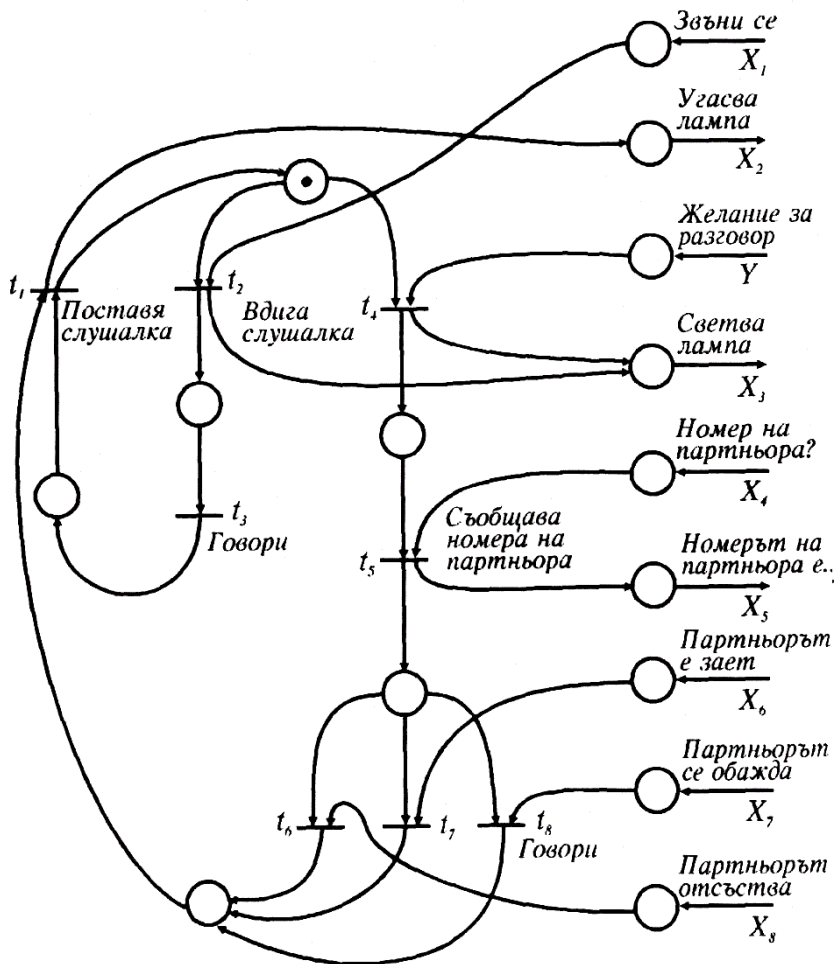
Решение:

Поведение на абонат i :

Полагания:

- t_1 - поставя слушалката;
- t_2 - вдига слушалката;
- t_3, t_8 - говори;
- t_4 - започва разговор;
- t_5 - съобщава номера на партньора;
- t_6 - благодари;

- t_7 - благодари;
 x_1 - звъни се;
 x_2 - загасва лампа;
 x_3 - светва лампа;
 x_4 - номер на партньора?



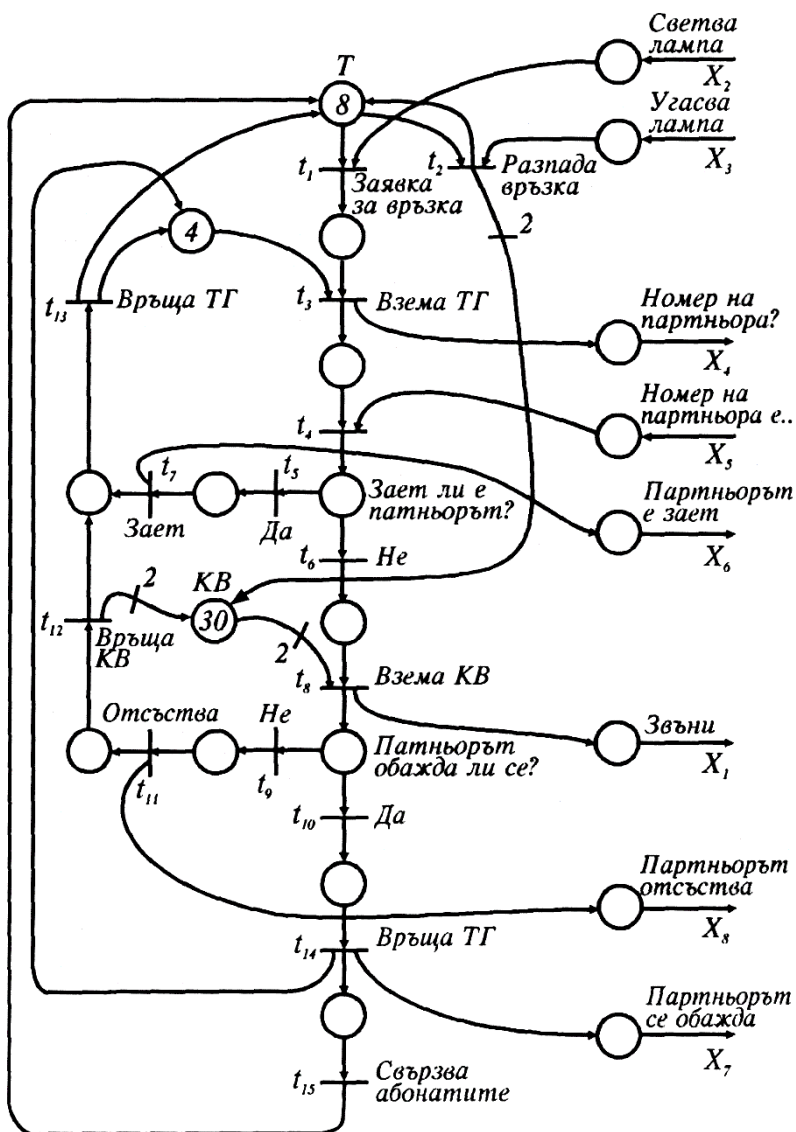
Фиг. 11.4. Граф на ОММП за абонат

x_5 - номер на партньора...
 x_6 - партньорът е зает;
 x_7 - партньорът се обажда;
 x_8 - партньорът отсъства;
 y - желание за разговор.

Поведение на телефонистките:

Полагания:

t_1 - заявка за връзка;
 t_2 - разпадане на връзката;
 t_3 - взема ТГ и пита за номер на партньора;
 t_4 - получава номер на партньора;
 t_5 - партньорът е зает;
 t_6 - партньорът не е зает;
 t_7 - съобщава, че партньора е зает;
 t_8 - взема КВ;
 t_9 - партньорът не се обажда;
 t_{10} - партньорът се обажда;
 t_{11} - съобщава, че партньора отсъства;
 t_{12} - връща КВ;
 t_{13} - връща ТГ;
 t_{14} - съобщава, че партньора се обажда;
 t_{15} - свързва абонатите.



Фиг. 11.5. Граф на ОММП за телефонистка

Упражнение 12.

Представяне на моделите в паметта.

Задача 1.

Даден е регулярен израз.

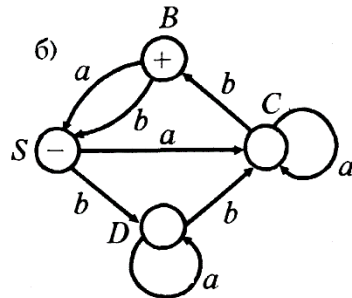
$$R = (a+ba^*b)a^*[b(a+b)(a+ba^*b)a^*]^*$$

Да се представи R чрез:

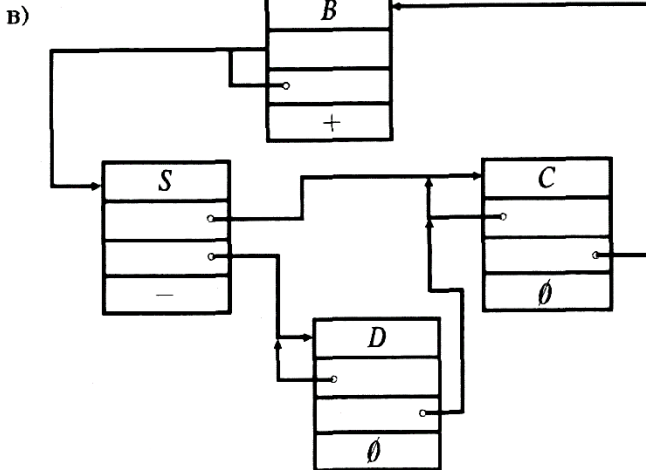
- регулярна граматика;
- краен автомат;
- списъчни структури в паметта.

Решение:

- $S \rightarrow aC;$
 $S \rightarrow bD;$
 $D \rightarrow aD;$
 $D \rightarrow bC;$
 $C \rightarrow aC;$
 $C \rightarrow bB;$
 $B \rightarrow aS;$
 $B \rightarrow bS;$
 $B \rightarrow \wedge$

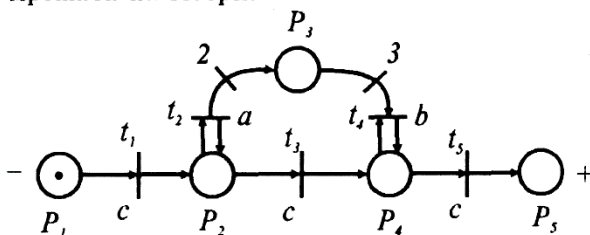


Фиг. 12.1. Краен автомат



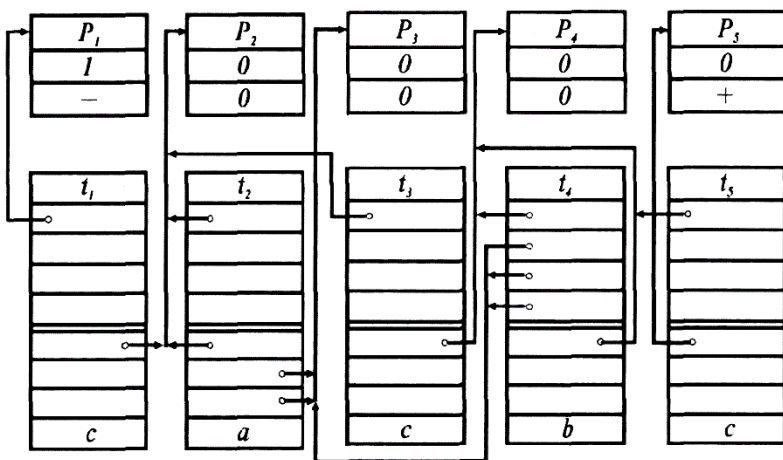
Фиг. 12.2. Списъчна структура

Задача 2. Да се представи като списъчна структура в паметта мрежата на Петри:



Фиг. 12.3. Граф на ОММП

Решение:



Фиг. 12.4. Списъчна структура

Задача 3. Езикът L е създаден чрез граматиката

$$G = \{ \Sigma, N, P, S \}.$$

Да се представи чрез:

- а) регулярен израз;
- б) краен автомат;
- в) списъчни структури.

$$\Sigma = \{a, b\}, \quad N = \{S, A, B, C\}$$

$$P = \{p_i \mid i = 1, 9\}; \quad S = \{S\}$$

$$p_1: S \rightarrow aB;$$

$$p_6: C \rightarrow bC;$$

$$p_2: S \rightarrow bA;$$

$$p_7: C \rightarrow aB;$$

$$p_3: S \rightarrow \wedge;$$

$$p_8: B \rightarrow aA;$$

$$p_4: A \rightarrow aC;$$

$$p_9: B \rightarrow bS;$$

$$p_5: A \rightarrow bC;$$

Решение:

а) преобразува се граматиката в регулярен израз;

$$C = b^*aB;$$

$$A = (a + b)C = (a + b)b^*aB;$$

$$B = aA + bS = a(a + b)b^*aB + bS;$$

$$B = (a(a + b)b^*a)^*bS; \quad (\text{правило на Ардън})$$

$$S = aB + bA + \wedge = aB + b(a + b)b^*aB + \wedge =$$

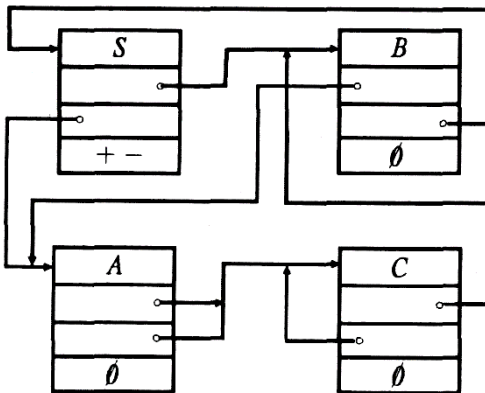
$$= (a + b(a + b)b^*a)B + \wedge =$$

$$= (a + b(a + b)b^*a)(a(a + b)b^*a)^*bS + \wedge =$$

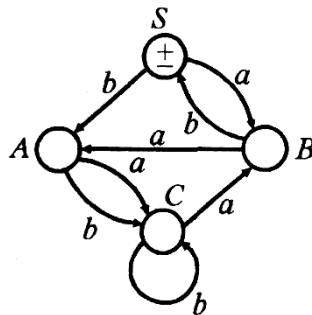
$$= ((a + b(a + b)b^*a)(a(a + b)b^*a)^*b)^*$$

(правило на Ардън)

б) преобразуване в краен автомат;



Фиг. 12.6. Списъчна структура



Фиг. 12.5. Краен автомат

в) представяне
чрез списъчни
структури.

Задача 4. Езикът L е зададен чрез граматиката G .

$$G = \{\Sigma, N, P, S\};$$

$$\Sigma = \{a, b, c\}, N = \{S, A, B, D\}; P = \{p_i | i = 1, 7\}, S = \{S\}$$

$$p_1: S \rightarrow Ac;$$

$$p_5: D \rightarrow bD;$$

$$p_2: A \rightarrow BAD;$$

$$p_6: D \rightarrow ba;$$

$$p_3: B \rightarrow aB;$$

$$p_7: A \rightarrow c.$$

$$p_4: B \rightarrow ab;$$

Да се представи L чрез:

- израз;
- ОММП;
- списъчни структури.

Решение:

а) Образуване на израз:

$$A = B^n A D^n = B^n c D^n; \quad \text{за } \forall n \geq 0;$$

$$D = b^k a; \quad \text{за } \forall k > 0;$$

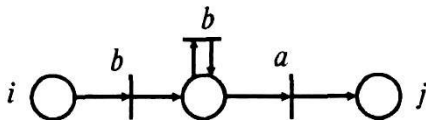
$$B = a^l b; \quad \text{за } \forall l > 0;$$

$$A = (a^l b)^n c (b^k a)^n; S = (a^l b)^n c (b^k a)^n c;$$

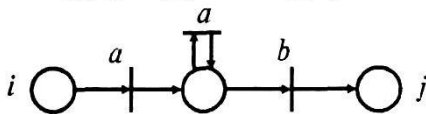
$$L(C) = \{(a^l b)^n c (b^k a)^n c | k, l > 0, n \geq 0\};$$

б) Съставя се ОММП:

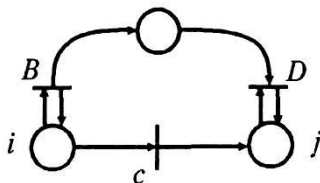
За $D = b^k a$;



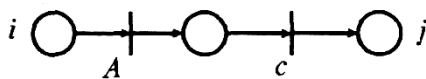
За $B = a^l b$ $l > 0$



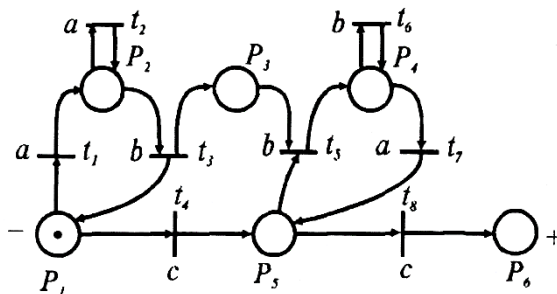
За $A = B^n c D^n$



За $S = Ac$

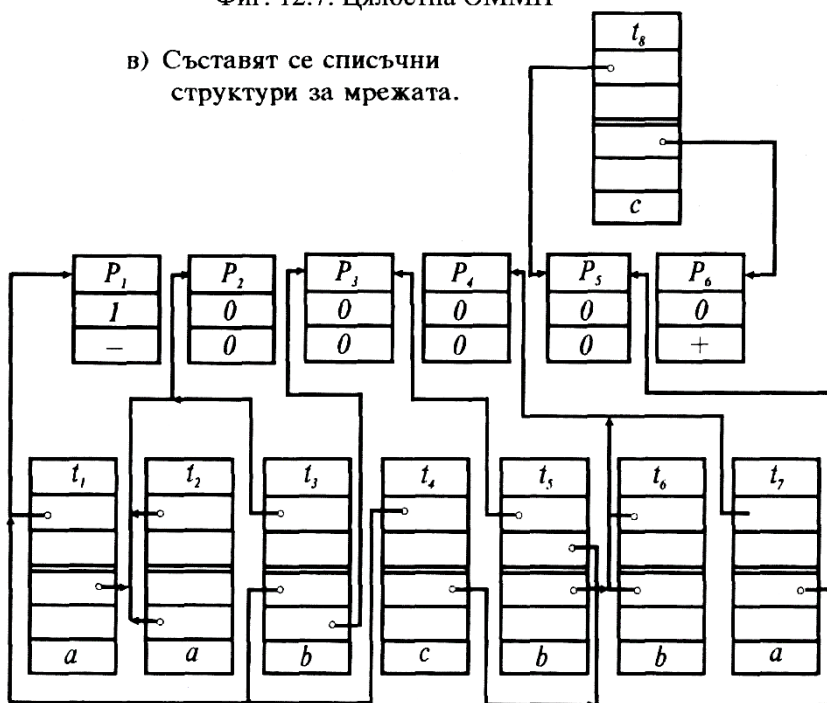


Елементите се свързват в обща мрежа S .



Фиг. 12.7. Цялостна ОММП

в) Съставят се списъчни структури за мрежата.



Фиг. 12.8. Списъчна структура

Упражнение 13.

Препозиционна (съждителна) логика.
Изчисления в препозиционната логика.

Определения

Съждение: Елементарно твърдение с известна истинност, която се определя от човека и не подлежи на по-нататъшен анализ бележат се с A, B, C като смисълът се игнорира.

Съждителна логика: Занимава се с методите за установяване истинността на сложни съждения (логически изрази), независимо от истинността на елементарните съждения.

Език на съждителната логика.

$$\Sigma = \{G1 \cup G2 \cup G3 \mid G1 = \{A_0, A_1, \dots, A_n\}, \\ G2 = \{^-, \wedge, \vee, \rightarrow, \leftrightarrow\}, G3 = \{(\cdot)\},$$

където: $G1$ -множество от съждителни променливи;

$G2$ -множество от логически съюзи;

$G3$ -множество от спомагателни символи.

Формули в съждителната логика (ППФ)

Синтактично правилно построени логически изрази (формули) ППФ:

Образуват се по следните правила:

1. Съждителните променливи са ППФ.

2. Ако P и Q са ППФ,

то $\bar{P}, (P \wedge Q), (P \vee Q), (P \rightarrow Q), (P \leftrightarrow Q)$ са ППФ

Преобразуване на ППФ

$$T \rightarrow B \Leftrightarrow B \quad B \rightarrow T \Leftrightarrow T$$

$$B \rightarrow B \Leftrightarrow T \quad \bar{B} \rightarrow B \Leftrightarrow B$$

$$T \wedge B \Leftrightarrow B \quad B \wedge B \Leftrightarrow B \quad F \wedge B \Leftrightarrow F$$

$$T \leftrightarrow B \Leftrightarrow B \quad B \leftrightarrow T \Leftrightarrow B \quad F \leftrightarrow B \Leftrightarrow \bar{B} \quad B \leftrightarrow F \Leftrightarrow \bar{B}$$

$$T \vee B \Leftrightarrow T \quad B \vee B \Leftrightarrow B \quad F \vee B \Leftrightarrow B$$

$$B \leftrightarrow B \Leftrightarrow T \quad B \leftrightarrow \bar{B} \Leftrightarrow F \quad \bar{B} \leftrightarrow B \Leftrightarrow F$$

Формули за две променливи

$$A \rightarrow B \Leftrightarrow \overline{A} \vee B \quad A \rightarrow B \Leftrightarrow \overline{(A \wedge \overline{B})}$$

$$A \leftrightarrow B \Leftrightarrow (A \wedge B) \vee (\overline{A} \wedge \overline{B}) \Leftrightarrow A \rightarrow B \vee B \rightarrow A \\ \Leftrightarrow (\overline{A} \vee B) \wedge (B \vee \overline{A})$$

$$\overline{(A \leftrightarrow B)} \Leftrightarrow \overline{((A \rightarrow B) \wedge (B \rightarrow A))} \Leftrightarrow \\ \overline{(A \rightarrow B)} \vee \overline{(B \rightarrow A)} \Leftrightarrow (A \wedge \overline{B}) \vee (B \wedge \overline{A})$$

Закони на Де Морган

$$\overline{(A \vee B)} \Leftrightarrow \overline{A} \wedge \overline{B} \quad \overline{(A \wedge B)} \Leftrightarrow \overline{A} \vee \overline{B}$$

Дистрибутивни закони

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

Комутативни закони

$$A \leftrightarrow B \Leftrightarrow B \leftrightarrow A$$

Асоциативни закони

$$(A \leftrightarrow B) \leftrightarrow C \Leftrightarrow A \leftrightarrow (B \leftrightarrow C)$$

Аксиоми за предпоставките:

- аксиома за предпоставката

$$\Gamma, A \Rightarrow A$$

- въвеждане на предпоставка

$$\frac{\Gamma \Rightarrow B}{\Gamma, A \Rightarrow B}$$

- елиминиране на предпоставката

$$\frac{(\Gamma, A \Rightarrow B) \wedge (\Gamma, \overline{A} \Rightarrow B)}{\Gamma \Rightarrow B}$$

$$\Gamma \Rightarrow T$$

$$\Gamma \Rightarrow \overline{F}$$

$$A \Rightarrow A$$

Правила за логическите съюзи:

За $\vee$

- въвеждане

$$\frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A \vee B}$$
$$\frac{\Gamma \Rightarrow B}{\Gamma \Rightarrow A \vee B}$$

- елиминирание

$$\frac{(\Gamma, A \Rightarrow C) \wedge (\Gamma, B \Rightarrow C) \wedge (\Gamma \Rightarrow A \vee B)}{\Gamma \Rightarrow C}$$

За $\wedge$

- въвеждане

$$\frac{(\Gamma \Rightarrow A) \wedge (\Gamma \Rightarrow B)}{\Gamma \Rightarrow A \wedge B}$$

- елиминирание

$$\frac{\Gamma \Rightarrow A \wedge B}{\Gamma \Rightarrow A}; \quad \Gamma \Rightarrow \frac{A \wedge B}{\Gamma \Rightarrow B}$$

За $\rightarrow$

- въвеждане

$$\frac{\Gamma, A \rightarrow B}{\Gamma \Rightarrow A \rightarrow B}$$

- елиминирание:

$$\frac{(\Gamma \Rightarrow A) \wedge (\Gamma \Rightarrow A \rightarrow B)}{\Gamma \Rightarrow B}$$

За $\neg$

- въвеждане

$$\frac{(\Gamma, A \Rightarrow B) \wedge (\Gamma, A \Rightarrow \bar{B})}{\Gamma \Rightarrow \bar{A}}$$

- елиминирание

$$\frac{(\Gamma \Rightarrow A) \wedge (\Gamma \Rightarrow \bar{A})}{\Gamma \Rightarrow B}$$

За =

$$\frac{\Gamma, A}{\Gamma, A} ; \quad \frac{\Gamma, \bar{A}}{\Gamma, A}$$

За $\leftrightarrow$

- въвеждане

$$\frac{(\Gamma \Rightarrow A \rightarrow B) \wedge (\Gamma \Rightarrow B \rightarrow A)}{\Gamma \Rightarrow A \leftrightarrow B}$$

- елиминирание

$$\frac{\Gamma \Rightarrow A \leftrightarrow B}{\Gamma \Rightarrow A \rightarrow B} ; \quad \frac{\Gamma \Rightarrow A \leftrightarrow B}{\Gamma \Rightarrow B \rightarrow A}$$

Задача 1.

Да се докаже, че ППФ е общовалидна (еквивалентна) на T .

$$((P \wedge Q) \rightarrow R) \wedge (P \rightarrow Q) \rightarrow (P \rightarrow R)$$

Решение:

$$\begin{aligned} & \overline{\overline{(P \wedge Q) \vee R} \wedge (\bar{P} \vee Q) \vee (\bar{P} \vee R)} \\ & (P \wedge Q) \wedge \bar{R} \vee (P \wedge \bar{Q}) \vee (\bar{P} \vee R) \\ & P \wedge (Q \wedge \bar{R} \vee \bar{Q}) \vee \bar{P} \vee R \\ & \bar{P} \vee \bar{R} \vee \bar{Q} \vee R \Leftrightarrow T \end{aligned}$$

Задача 2.

Да се докаже общовалидността на ППФ.

$$((P \rightarrow Q) \wedge (P \vee R)) \rightarrow (Q \vee R)$$

Решение:

$$\begin{aligned} & \overline{((\overline{P} \vee Q) \wedge (P \vee R)) \vee Q \vee R} \\ & (P \wedge \overline{Q}) \vee (\overline{P} \wedge \overline{R}) \vee Q \vee R \\ & P \vee (\overline{P} \wedge \overline{R}) \vee R \\ & \overline{R} \vee R \Leftrightarrow T \end{aligned}$$

Задача 3.

Да се докаже еквивалентността на следните ППФ.

$$(A \vee B) \wedge (A \vee C) \wedge (B \vee D) \wedge (C \vee D) \Leftrightarrow ((A \wedge D) \vee (B \wedge C))$$

Упътване:

Лява част. Прилагат се дистрибутивни закони:

$$\begin{aligned} & (A \vee (B \wedge C)) \wedge (D \vee (B \wedge C)) \Leftrightarrow \\ & ((A \wedge D) \vee (B \wedge C)) \end{aligned}$$

Задача 4.

Да се докаже, че множеството от логически съюзи:

$$A = \{\wedge, \vee, \leftrightarrow\} \text{ е непълно;}$$

$$B = \{\neg, \rightarrow\} \text{ е пълно.}$$

Доказателство:

$$\begin{aligned} B: X \wedge Y &= \overline{X \rightarrow \overline{Y}} \\ X \vee Y &= \overline{X} \rightarrow Y \\ X \leftrightarrow Y &= (X \rightarrow Y) \wedge (Y \rightarrow X) = \\ & \overline{(X \rightarrow Y) \rightarrow (Y \rightarrow X)} \\ A: \overline{X} &= X \rightarrow 0; \overline{X} = X \rightarrow F \end{aligned}$$

$F = 0$ не принадлежи на A . От променливи $X_1, X_2, \dots, X_n$ могат да се получат само функции, зависещи съществено от не повече от 1 променлива.

Задача 5.

Да се изведе правилно построената формула

$$\Rightarrow P \vee \bar{P}$$

Решение:

$P \Rightarrow P$	аксиома за предпоставката;
$P \Rightarrow P \vee \bar{P}$	въвеждане на $\vee$;
$\bar{P} \Rightarrow \bar{P}$	аксиома за предпоставката;
$\bar{P} \Rightarrow \bar{P} \vee P$	въвеждане $\vee$;
$\Rightarrow \bar{P} \vee P$	елиминирание предпоставката.

Задача 6.

Да се изведе ППФ:

$$\Rightarrow (p \wedge q \rightarrow r) \wedge (p \rightarrow q) \rightarrow (p \rightarrow r)$$

Решение:

1. $(p \wedge q \rightarrow r), (p \rightarrow q), p \Rightarrow p$ - аксиома за предпоставката
2. $(p \wedge q \rightarrow r), (p \rightarrow q), p \Rightarrow p \rightarrow q$ - аксиома за предпоставката
3. $(p \wedge q \rightarrow r), (p \rightarrow q), p \Rightarrow q$ - елиминирание на $\rightarrow$ ред 1 и 2
4. $(p \wedge q \rightarrow r), (p \rightarrow q), p \Rightarrow p \wedge q$ - въвеждане $\wedge$ ред 1,3
5. $(p \wedge q \rightarrow r), (p \rightarrow q), p \Rightarrow p \wedge q \rightarrow r$ - аксиома за предпоставката
6. $(p \wedge q \rightarrow r), (p \rightarrow q), p \Rightarrow r$ - елиминирание на $\rightarrow$ ред 4,5
7. $(p \wedge q \rightarrow r), (p \rightarrow q) \Rightarrow p \rightarrow r$ - въвеждане $\rightarrow$ ⁶
 $\Rightarrow (p \wedge q \rightarrow r) \wedge (p \rightarrow q) \rightarrow (p \rightarrow r)$ - въвеждане $\rightarrow$ ⁷

Задача 7.

Да се изведе ППФ:

$$\Rightarrow (p \rightarrow q) \leftrightarrow (\bar{q} \rightarrow \bar{p})$$

Решение:

- | | |
|--|---------------------------------------|
| 1. $p \rightarrow q, \bar{q}, p \Rightarrow p$ | аксиома |
| 2. $p \rightarrow q, \bar{q}, p \Rightarrow p \rightarrow q$ | аксиома |
| 3. $p \rightarrow q, \bar{q}, p \Rightarrow q$ | елиминираме $\rightarrow 1, 2$ |
| 4. $p \rightarrow q, \bar{q}, p \Rightarrow \bar{q}$ | аксиома |
| 5. $p \rightarrow q, \bar{q} \Rightarrow \bar{p}$ | ред 3, 4, въвеждане $\neg$ |
| 6. $p \rightarrow q \Rightarrow \bar{q} \rightarrow \bar{p}$ | въвеждане на $\rightarrow 5$ |
| 7. $\Rightarrow (p \rightarrow q) \rightarrow (\bar{q} \rightarrow \bar{p})$ | въвеждане на $\rightarrow 6$ |
| 8. $(\bar{q} \rightarrow \bar{p}), p, \bar{q} \Rightarrow \bar{q}$ | аксиома |
| 9. $(\bar{q} \rightarrow \bar{p}), p, \bar{q} \Rightarrow \bar{q} \rightarrow \bar{p}$ | аксиома |
| 10. $(\bar{q} \rightarrow \bar{p}), p, \bar{q} \Rightarrow \bar{p}$ | елиминираме $\rightarrow 8, 9$ |
| 11. $(\bar{q} \rightarrow \bar{p}), p, \bar{q} \Rightarrow p$ | аксиома |
| 12. $(\bar{q} \rightarrow \bar{p}), p \Rightarrow \bar{q}$ | въвеждане ред 10 и 11 |
| 13. $(\bar{q} \rightarrow \bar{p}), p \Rightarrow q$ | елиминираме = ред 12 |
| 14. $(\bar{q} \rightarrow \bar{p}) \Rightarrow p \rightarrow q$ | въвеждане $\rightarrow$ ред 13 |
| 15. $\Rightarrow (\bar{q} \rightarrow \bar{p}) \rightarrow (p \rightarrow q)$ | въвеждане $\rightarrow$ ред 14 |
| 16. $\Rightarrow (\bar{q} \rightarrow \bar{p}) \leftrightarrow (p \rightarrow q)$ | въвеждане $\leftrightarrow$ ред 7, 15 |

План за решение:

Извеждат се двете ППФ

$$\Rightarrow (p \rightarrow q) \rightarrow (\bar{q} \rightarrow \bar{p}) \text{ и}$$

$$\Rightarrow (\bar{q} \rightarrow \bar{p}) \rightarrow (p \rightarrow q) \text{ и се прилага правило}$$

то за еквивалентност. За извеждане на ППФ

$$\Rightarrow (p \rightarrow q) \rightarrow (\bar{q} \rightarrow \bar{p}) \text{ се извеждат}$$

$$(\bar{q} \rightarrow \bar{p}), p \Rightarrow q$$

и се въвежда правилото за $\rightarrow$ два пъти. За да се въведе p единственото правило е правилото за $\bar{}$. За да се приложи това правило, трябва да се намери такива ППФ B , че

$$(p \rightarrow q), \bar{q}, p \Rightarrow B \quad \text{и}$$

$$(p \rightarrow q), \bar{q}, p \Rightarrow \bar{B}.$$

Например B може да бъде q , следователно трябва да се изведе най-напред

$$(p \rightarrow q), \bar{q}, p \Rightarrow q$$

и след това

$$(p \rightarrow q), \bar{q}, p \Rightarrow \bar{q}.$$

Задача 8.

Да се изведе ППФ:

$$\Rightarrow ((p \rightarrow q) \wedge (p \vee r) \rightarrow (q \vee p))$$

Решение:

1. $(p \rightarrow q), (p \vee r), p \Rightarrow p$ аксиома за предпоставката
2. $(p \rightarrow q), (p \vee r), p \Rightarrow p \rightarrow q$ аксиома за предпоставката
3. $(p \rightarrow q), (p \vee r), p \Rightarrow q$ премахване $\rightarrow$
4. $(p \rightarrow q), (p \vee r), p \Rightarrow q \vee r$ въвеждане $\vee$
5. $(p \rightarrow q), (p \vee r), r \Rightarrow r$ аксиома за предпоставката
6. $(p \rightarrow q), (p \vee r), r \Rightarrow q \vee r$ въвеждане $\vee$
7. $(p \rightarrow q), (p \vee r) \Rightarrow p \vee r$ аксиома за предпоставката

$$\begin{aligned}
 &8. (p \rightarrow q), (p \vee r) \Rightarrow q \vee r && \text{премахване } \vee \\
 &\Rightarrow ((p \rightarrow q) \wedge (p \vee r) \rightarrow (q \vee p)) && \text{въвеждане } \rightarrow
 \end{aligned}$$

Задача 9.

Да се изведе ППФ:

$$\Rightarrow \overline{(A \vee B) \rightarrow C \vee A \rightarrow B} \vee (A \rightarrow C)$$

Решение:

$$\begin{aligned}
 &(A \vee B) \rightarrow C, A \Rightarrow A && \text{аксиома за} \\
 & && \text{предпоставката} \\
 &(A \vee B) \rightarrow C, A \Rightarrow A \vee B && \text{правило за } \vee \\
 &(A \vee B) \rightarrow C, A \Rightarrow A \vee B \rightarrow C && \text{аксиома за} \\
 & && \text{предпоставката} \\
 &(A \vee B) \rightarrow C, A \Rightarrow C && \text{правило за } \rightarrow \\
 &(A \vee B) \rightarrow C \Rightarrow A \rightarrow C && \text{правило за } \rightarrow \\
 &(A \vee B) \rightarrow C \Rightarrow \overline{(A \vee B) \rightarrow C} \vee \overline{A \rightarrow B} \vee A \rightarrow C && \text{правило за } \vee \\
 & && \text{правило за } \vee \\
 &\overline{(A \vee B) \rightarrow C} \Rightarrow \overline{(A \vee B) \rightarrow C} && \text{аксиома за} \\
 & && \text{предпоставката} \\
 &\overline{(A \vee B) \rightarrow C} \Rightarrow \overline{(A \vee B) \rightarrow C} \vee \overline{A \rightarrow B} \vee A \rightarrow C && \text{правило за } \vee \\
 & && \text{правило за } \vee \\
 &\Rightarrow \overline{(A \vee B) \rightarrow C} \vee \overline{A \rightarrow B} \vee A \rightarrow C && \text{правило за} \\
 & && \text{елиминирание} \\
 & && \text{на предпоставка}
 \end{aligned}$$

Упражнение 14.

Предикатна логика. Метод на резолюцията.

Задача 1.

Дадена е ППФ A :

$$A : ((\overline{p_1(a)} \vee \overline{p_2(b)} \vee p_3(a)) \wedge (\overline{p_1(a)} \vee p_2(b)) \wedge p_1(a) \wedge \overline{p_3(a)})$$

в клаузна форма. Да се провери за неудовлетвореност чрез метода на резолюцията.

Решение:

Базови клаузи:

$$1) \overline{p_1(a)} \vee \overline{p_2(b)} \vee p_3(a)$$

$$2) \overline{p_1(a)} \vee \overline{p_2(b)}$$

$$3) p_1(a)$$

$$4) \overline{p_3(a)}$$

$$5) \overline{p_1(a)} \vee p_3(a) \quad - \text{резолвента от 1, 2}$$

$$6) p_3(a) \quad - 3, 5$$

$$7) \square \quad - 4, 6$$

Задача 2. Дадена е ППФ A :

$$A : \exists y \forall z (P(z, y) \leftrightarrow \exists x (P(z, x) \wedge P(x, z)))$$

Да се преобразува в клаузна форма $\overline{A}$

$$\forall x_1, \forall x_2, \dots, \forall x_n (c_1, c_2, \dots, c_n)$$

Решение:

Етап 1: Елиминирание на $\leftrightarrow$:

$$A \leftrightarrow B \Leftrightarrow (\overline{A} \vee B) \wedge (A \vee \overline{B})$$

$$\begin{aligned}\overline{A} : \exists y \forall z (p(z, y) &\leftrightarrow \overline{\exists x (p(z, x) \wedge p(x, z))}) \\ \overline{A} : \exists y \forall z \{ &\overline{[p(y, z) \vee \exists x (p(z, x) \wedge p(x, z))]} \wedge \\ &[p(z, y) \vee \exists x (p(z, x) \wedge p(x, z))]\} \end{aligned}$$

Етап 2: Разчленяване на инверсията (използват се законите на Де Морган и правило за кванторите):

$$\begin{aligned}\overline{\exists x p(x)} &\equiv \forall x \overline{p(x)} \\ \overline{A} : \exists y \forall x \{ &\overline{[p(z, y) \vee \forall x (\overline{p(z, x)} \vee \overline{p(x, z)})]} \wedge \\ &[p(z, y) \vee \exists x (p(z, x) \wedge p(x, z))]\} \end{aligned}$$

Етап 3: Елиминирание на кванторите за съществуване:

$$\begin{aligned}\exists x (\dots x) &\equiv (f(z)) & \exists y (\dots y) &\equiv (\dots a) \\ \overline{A} : \forall z \{ &\overline{[p(z, a) \vee \forall x (\overline{p(z, x)} \vee \overline{p(x, z)})]} \wedge \\ &[p(z, a) \vee p(z, f(z)) \wedge p(f(z), z)]\} \end{aligned}$$

Етап 4: Придвижване на $\forall$ наляво

$$\begin{aligned}\overline{A} : \forall z \forall x \{ &\overline{[p(z, a) \vee \overline{p(z, x)} \vee \overline{p(x, z)}]} \wedge \\ &[(p(z, a) \vee p(z, f(z))) \wedge p(f(z), z)]\} \end{aligned}$$

Етап 5: Дистрибутивни закони:

$$\begin{aligned}A \vee (B \wedge C) &\Leftrightarrow (A \vee B) \wedge (A \vee C) \\ A \wedge (B \vee C) &\Leftrightarrow (A \wedge B) \vee (A \wedge C) \\ \overline{A} : \forall z \forall x \{ &\overline{[p(z, a) \vee \overline{p(z, x)} \vee \overline{p(x, z)}]} \wedge \\ &[(p(z, a) \vee p(z, f(z))) \wedge (p(z, f(z)) \vee p(f(z), z))]\} \end{aligned}$$

Етап 6: Формиране на клаузи

- 1) $\overline{p(z, a) \vee \overline{p(z, x)} \vee \overline{p(x, z)}}$
- 2) $p(z, a) \vee p(z, f(z))$

$$3) p(z, f(z)) \vee p(f(z), z)$$

Да се приложи базовия метод на резолюцията при:

$$z = \{a\}, x = \{a, b\}, f(a) = b$$

$$1) \overline{p(a, a)} \vee \overline{p(a, a)} \vee \overline{p(a, a)} \leftrightarrow \overline{p(a, a)}$$

$$2) \overline{p(a, a)} \vee \overline{p(a, b)} \vee \overline{p(b, a)}$$

$$3) p(a, a) \vee p(a, b)$$

$$4) p(a, a) \vee p(b, a)$$

$$5) \overline{p(b, a)} - \text{резолвента от 2, 3}$$

$$6) p(a, a) - \text{от 4, 5}$$

$$7) \square - \text{от 1, 6}$$

Задача 3. Да се запише определението за сестри като ППФ и да се преобразува в клаузна форма. Да се използват предикатите.

$q(x)$	- x е жена
$r(x, z)$	- z е майка на x
$s(x, r)$	- r е баща на x
$p(x, y)$	- x и y са сестри

Определение: x и y са сестри, ако са жени и имат обща майка или общ баща.

Решение:

$$A: \forall x \forall y \{ [q(x) \wedge q(y) \wedge (\exists z [r(x, z) \wedge r(y, z)]) \vee \exists w [s(x, w) \wedge s(y, w)]] \rightarrow p(x, y) \}$$

Етап 1. Елиминирание на $\rightarrow$

Използват се правилата:

$$(A \rightarrow B \equiv \overline{A} \vee B)$$

$$\forall x \forall y ([\overline{q(x)} \vee \overline{q(y)} \vee \overline{C} \vee p(x, y)]),$$

където:

$$C = (\exists z [r(x, z) \wedge r(y, z)] \vee \exists w [s(x, w) \wedge s(y, w)])$$

Етап 2. Разчленяване на инверсията:

$$C: \overline{\exists z[r(x, z) \wedge r(y, z)] \vee \exists w[s(x, w) \wedge s(y, w)]}$$

$$C: \overline{\forall z[r(x, z) \wedge r(y, z)] \wedge \forall w[s(x, w) \wedge s(y, w)]}$$

$$C: \overline{\forall z[r(x, z) \vee r(y, z)] \wedge \forall w[s(x, w) \vee s(y, w)]}$$

Етап 3. Елиминиране на $\exists$ - (не се прилагат, защото няма);

Етап 4. Изместване кванторите $\forall$ наляво:

$$\forall x \forall y \forall z \forall w \{ \overline{q(x)} \vee \overline{q(y)} \vee [(\overline{r(x, z)} \vee \overline{r(y, z)}) \wedge (\overline{s(x, w)} \vee \overline{s(y, w)})] \vee p(x, y) \}$$

Етап 5. Дистрибутивни закони.

$$[\overline{q(x)} \vee \overline{q(y)} \vee p(x, y) \vee \overline{r(x, z)} \vee \overline{r(y, z)}] \wedge [\overline{q(x)} \vee \overline{q(y)} \vee p(x, y) \vee \overline{s(x, w)} \vee \overline{s(y, w)}]$$

Етап 6. Формиране на клаузите:

$$1) \overline{q(x)} \vee \overline{q(y)} \vee \overline{r(x, z)} \vee \overline{r(y, z)} \vee p(x, y)$$

$$2) \overline{q(x)} \vee \overline{q(y)} \vee \overline{s(x, w)} \vee \overline{s(y, w)} \vee p(x, y)$$

Етап 7. Образуване програма на Пролог:

$$1) p(x, y) : -q(x), q(y), r(x, w), r(y, w)$$

$$2) p(x, y) : -q(x), q(y), s(x, w), s(y, w)$$

Заместване на полаганията:

сестри(X, Y) :-жена(X), жена(Y), майка(W, X), майка(W, Y).

сестри(X, Y) :-жена(X), жена(Y), баща(W, X), баща(W, Y).

Упражнение 15.

Общ метод на резолюцията. Език за логическо програмиране ПРОЛОГ

Задача 1.

Да се формулират фактите и въпроса, дадени в следната загадка като правилно построена формула от предикатна логика. Да се използва общият метод на резолюцията за да се отговори на зададения въпрос.

Загадка: Тони, Майк и Джон са членове на алпийски клуб. Всеки член на алпийски клуб е или скиор, или алпинист, или скиор и алпинист. Никой алпинист не обича дъжд, а всеки скиор обича сняг. Майк обича това, което Тони мрази и мрази онова, което Тони обича. Тони обича дъжд и сняг.

Въпрос: Има ли член на алпийския клуб, който е алпинист, но не е скиор?

Решение: За опростяване на израза се правят следните обозначения:

$ac(x)$	- X е член на алпийския клуб
$ap(x)$	- X е алпинист
$sk(x)$	- X е скиор
$l(X, Y)$	- X обича Y

и константи:

a - Тони	d - дъжд
b - Майк	s - сняг
c - Джон	

Правилно построената формула на загадката се съставя по условието на загадката, като формализираните изречения се свързват с логическо и ($\wedge$):

$$\begin{aligned} & A: ac(a) \wedge ac(b) \wedge ac(c) \wedge \\ & \forall x[ac(x) \rightarrow sk(x) \vee ap(x) \vee (sk(x) \wedge ap(x))] \wedge \\ & \forall y[ap(y) \rightarrow \overline{l(y, d)}] \vee \forall z[sk(z) \rightarrow l(z, s)] \wedge \\ & \forall v[l(b, v) \rightarrow \overline{l(a, v)}] \wedge \forall w[\overline{l(b, w)} \rightarrow l(a, w)] \wedge \\ & l(a, d) \wedge l(a, s) \end{aligned}$$

Въпросът се представя чрез клаузата:

$$B: ac(x) \wedge ap(x) \wedge \overline{sk(x)}$$

Решение:

ППФ се преобразува в клаузна форма:

$$\forall x_1 \forall x_2 \dots \forall x_n (c_1 \wedge c_2 \wedge \dots \wedge c_n),$$

където $\forall x_1 \forall x_2 \dots \forall x_n$ - квантифицирани с квантор за общност променливи, участващи в ППФ (префикс) $c_1, c_2, \dots$ - клаузи на ППФ (матрица).

Резултантната клаузна форма е следната:

$$\begin{aligned} & \forall x \forall y \forall z \forall v \forall w \{ \overline{ac(a)} \wedge \overline{ac(b)} \wedge \overline{ac(c)} \wedge \\ & \overline{[ac(x) \vee sk(x) \vee ap(x)]} \wedge \overline{[ap(y) \vee l(y, d)]} \wedge \\ & \overline{[sk(z) \vee l(z, s)]} \wedge \overline{[l(b, v) \vee l(a, v)]} \wedge \\ & \overline{[l(b, w) \rightarrow l(a, w)]} \wedge l(a, d) \wedge l(a, s) \} \end{aligned}$$

Клаузата въпрос се инвертира:

$$B: \overline{ac(x)} \vee \overline{ap(x)} \vee sk(x)$$

За решаване на загатката се търси противоречие между аксиомите на проблема и клаузата въпрос (дефиницията на загатката). Противоречието между тях се търси по общия метод на резолюцията с използване на унифициращ алгоритъм. Ако съществува противоречие (празна клауза) в резултат от метода на резолюцията, отговорът на загатката е положителен. В противен случай - отрицателен.

$c_1: ac(a)$	$c_6: \overline{sk(z)} \vee l(z, s)$
$c_2: ac(b)$	$c_7: \overline{l(b, v)} \vee \overline{l(a, v)}$
$c_3: ac(c)$	$c_8: l(b, w) \wedge l(a, w)$
$c_4: \overline{ac(x_1)} \vee \overline{sk(x_1)} \vee \overline{ap(x_1)}$	$c_9: l(a, d)$
$c_5: \overline{ap(y)} \vee \overline{l(y, d)}$	$c_{10}: l(a, s)$

$$\overline{B}: \overline{ac(x_2)} \vee \overline{ap(x_2)} \vee sk(x_2)$$

Отправна точка на метода е клаузата $\overline{B}$. Търси се резолвента на тази клауза с клаузите на загадката по някаква стратегия за търсене. Простата стратегия на търсене вградена в езика Пролог открива първата възможна резолвента между $\overline{B}$ и c_1 .

Образува се резолвента между $\overline{B}$ и c_1 :

$c_{11}: \overline{ap(b)} \vee sk(b)$	$\overline{B} \text{ с } c_2 \text{ е унификатор } \langle x, b \rangle$
$c_{12}: \overline{ac(b)} \vee sk(b)$	$c_{11} \text{ с } c_4 \text{ с унификатор } \langle x, b \rangle$
$c_{13}: sk(b)$	$c_{12} \text{ с } c_2 \text{ с унификатор } \langle \rangle$
$c_{14}: l(b, s)$	$c_{13} \text{ с } c_6 \text{ с унификатор } \langle z, b \rangle$
$c_{15}: l(a, s)$	$c_{14} \text{ с } c_7 \text{ с унификатор } \langle v, s \rangle$
$c_{16}: \square$	$c_{15} \text{ с } c_{10}$

В резултат от извода отговорът на загадката е положителен. Допълнителната информация, получена чрез унификаторите, определя кой член на алпийския клуб отговаря на тези условия. Това е b (Майк). Освен тази информация, унификаторите съдържат и сведения за условията при които е направен изводът: например $\langle z, b \rangle$ определя Майк като скиор, $\langle v, s \rangle$ определя причината за това и т.н.

Задача 2. Да се докаже общовалидността на ППФ A чрез общия метод на резолюцията.

$$A: \exists x \exists y \forall z \{ [p(x, y) \rightarrow (p(y, z) \wedge p(z, z))] \wedge \\ [[p(x, y) \wedge q(x, y)] \rightarrow [q(x, z) \wedge q(z, z)]] \}$$

Необходимо е инвертиране на ППФ в инверсна форма и доказателство, че тя е неудовлетворима. За да се докаже неудовлетворимостта, формулата се преобразува в клаузна форма, върху която се прилага общия метод на резолюцията.

Преобразуване в клаузна форма:

Етап 1. Елиминирание импликацията.

$$\frac{\frac{\frac{}{\exists x \exists y \forall z \{ [\overline{p(x, y)} \vee (p(y, z) \wedge p(z, z))] \wedge$$

$$A : \frac{[\overline{p(x, y) \wedge q(x, y)}] \vee [q(x, z) \wedge q(z, z)]]}{}}$$

Етап 2. Разчленяване на инверсията.

$$\forall x \forall y \exists z \{ [p(x, y) \wedge (\overline{p(y, z)} \vee \overline{p(z, z)})] \vee [[p(x, y) \wedge q(x, y)] \wedge$$

$$\wedge [\overline{q(x, z)} \vee \overline{q(z, z)}]] \}$$

Етап 3. Елиминирание на $\exists z$:

$$A : \forall x \forall y \{ [p(x, y) \wedge (\overline{p(y, f(x, y))} \vee \overline{p(f(x, y), f(x, y))})] \vee$$

$$[[p(x, y) \wedge q(x, y)] \wedge [\overline{q(x, f(x, y))} \vee \overline{q(f(x, y), f(x, y))}]] \}$$

Полагания:

$$A \equiv p(x, y),$$

$$B \equiv p(y, f(x, y)),$$

$$C \equiv p(f(x, y), f(x, y)),$$

$$D \equiv q(x, y),$$

$$E \equiv \overline{q(x, f(x, y))},$$

$$F \equiv \overline{q(f(x, y), f(x, y))}$$

Етап 4. Кванторите са изместени

Етап 5. Дистрибутивни закони.

$$A \vee (B \wedge C) \Leftrightarrow (A \vee B) \wedge (A \vee C);$$

$$A \wedge (B \vee C) \Leftrightarrow (A \wedge C) \vee (B \wedge C)$$

Етап 6. Формиране на клаузи:

$$(A \vee A) \wedge (A \vee D) \wedge (A \vee E \vee F) \wedge (B \vee C \vee A) \wedge$$

$$(B \vee C \vee D) \wedge (B \vee C \vee E \vee F)$$

Клаузите са:

$$p(x_1, y_1);$$

$$\overline{p(y_2, f(x_2, y_2))} \vee \overline{p(f(x_2, y_2), f(x_2, y_2))} \vee q(x_2, y_2)$$

$$\overline{p(y_3, f(x_3, y_3))} \vee \overline{p(f(x_3, y_3), f(x_3, y_3))} \vee \overline{q(x_3, f(x_3, y_3))} \vee \\ \vee \overline{p(f(x_3, y_3), f(x_3, y_3))}$$

Прилага се общият метод на резолюцията.

$$1) p(x_1, y_1)$$

$$2) \overline{p(y_2, f(x_2, y_2))} \vee \overline{p(f(x_2, y_2), f(x_2, y_2))} \vee q(x_2, y_2)$$

$$3) \overline{p(y_3, f(x_3, y_3))} \vee \overline{p(f(x_3, y_3), f(x_3, y_3))} \vee \overline{q(x_3, f(x_3, y_3))}$$

$$\vee \overline{p(f(x_3, y_3), f(x_3, y_3))}$$

Резолвенти:

$$4) \overline{p(f(x_2, y_2), f(x_2, y_2))} \vee q(x_2, y_2)$$

унификатор: $\{<x_1, y_2>, <y_1, f(x_2, y_2)>\}$

$$5) q(x_2, y_2)$$

унификатор: $\{<x_1, f(x_2, y_2)>, <y_1, f(x_2, y_2)>\}$

6)

$$\overline{p(y_3, f(x_3, y_3))} \vee \overline{p(f(x_3, y_3), f(x_3, y_3))} \vee \overline{q(f(x_3, y_3), f(x_3, y_3))}$$

унификатор: $\{<x_2, f(x_3, y_3)>, <y_2, f(x_3, y_3)>\}$

- 7) $\overline{p(y_3, f(x_3, y_3))} \vee \overline{p(f(x_3, y_3), f(x_3, y_3))}$
унификатор: $\{<x_2, y_3>, <y_2, f(x_3, y_3)>\}$
- 8) $\overline{p(f(x_3, y_3), f(x_3, y_3))}$
унификатор: $\{<x_1, y_3>, <y_1, f(x_3, y_3)>\}$
- 9) $\square$ унификатор: $\{<x_1, f(x_3, y_3)>, <y_1, f(x_3, y_3)>\}$

Най-общия унификатор е:

$$\{<x, y>, <y, f(x, y)>\}$$

Задача 3. Дадени са предикатите:

сестра(x, y)	X и Y са сестри
брат(x, y)	X и Y са братя
майка(x, y)	X е майка на Y
баща(x, y)	X е баща на Y
съпруг(x, y)	X е съпруг на Y
мъж(x)	X е мъж
жена(x)	X е жена

Да се образуват програми на ПРОЛОГ за:

- а) чичо(x, y) - X е чичо на Y ;
б) внук(x, y) - X е внук на Y ;
в) братовчед(x, y) - X и Y са първи братовчеди.

а) Упътване:

Всеки x е чичо на y ако съществува z , който е брат на x и баща на y .

б) Отговор:

- чичо(x, y):- брат(x, z), баща(z, y).
внук(x, y):- мъж(x), родител(y, z), родител(z, x).
родител(x, y):- баща(x, y).
родител(x, y):- майка(x, y).

ЛИТЕРАТУРА

1. Агафонов, В. *Логическое программирование*, Мир, Москва, 1988.
2. Денев, Й. *Математическа лингвистика, Наука и изкуство*, София, 1980.
3. Кросли, Д. *Що е математическа логика, Наука и изкуство*, София, 1980.
4. Лавров, И. *Задачи по теории множеств, математической логике и теории алгоритмов*, Наука, Москва, 1984.
5. Линднер, Р. *Теория и практика структурного программирования*, Мир, Москва, 1982.
6. Манна, Зоар *Математическа теория на информацията, Наука и изкуство*, София, 1985.
7. Никольская, И. *Математическая логика, Высшая школа*, Москва, 1981.
8. Олунгрен, А. *Определение языков программирования интерпретирующими автоматами*, Мир, Москва, 1982.
9. Радославов, И. *Математическа лингвистика, Народна просвета*, София, 1982.
10. Рейуорд, В. *Теория формальных языков, Радио и связь*, Москва, 1988.
11. Стойчев, Л., А. Антонов. *Програмни езици за изкуствен интелект*, Техника, София, 1989.
12. Яблонский, С. *Введение в дискретную математику*, Наука, Москва, 1986.
13. Якоба, К. *Машина Тьюринга и рекурсивные функции*, Мир, Москва, 1980.
14. Янков Б. *Транслатори и операционни системи, ВМЕИ*, София, 1984.
15. Peterson, J. *Petri net theory and the modeling of systems*, Prentice Hall, 1981.

СЪДЪРЖАНИЕ

Упражнение 1. Множества. Операции с множества.....	4
Упражнение 2. Декартово произведение на множества. Отношения и функции в множествата	10
Упражнение 3. Изчисления с низове. Операции с низове. Регулярни множества и изрази	17
Упражнение 4. Формални граматики и езици	24
Упражнение 5. Крайни автомати - анализ	32
Упражнение 6. Крайни автомати. Синтез	40
Упражнение 7. Машини на Тюринг	46
Упражнение 8. Мрежи на Петри. Графи на мрежи на Петри. Анализ и синтез	51
Упражнение 9. Маркирани мрежи на Петри. Действие на мрежите. Свойства на маркираните мрежи на Петри	55
Упражнение 10. Мрежови езици	60
Упражнение 11. Моделиране на паралелни процеси	64
Упражнение 12. Представяне на моделите в паметта	70
Упражнение 13. Препозиционна (съждителна) логика. Изчисления в препозиционната логика	75
Упражнение 14. Предикатна логика. Метод на резолуцията	84
Упражнение 15. Общ метод на резолуцията. Език за логическо програмиране ПРОЛОГ	88
Литература	94

ДИСКРЕТНИ СТРУКТУРИ

Ръководство за семинарни упражнения

Автори:

доц. д-р инж. Анатолий Стефанов Антонов

доц. д-р инж. Владимир Николов Николов

ас. д-р инж. Ивайло Пламенов Пенев

Рецензент:

доц. д-р инж. Трифон Русков

Предпечатна подготовка: "Геа-принт" Варна

Печат: ТУ-Варна