|  |  |
| --- | --- |
| **UNIVERITATEA DE STAT DIN MOLDOVA**  **Facultatea de Matematică şi Informatică**  **Departamentul de Informatică** | |
|  | **APROB**  **\_\_\_ \_\_\_\_\_\_\_\_\_\_\_2021**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Director Departament**  **V. Ungureanu, dr., conf. univ.** |

**TEST Nr. 14**

**pentru examenul scris la disciplina *Inteligenţa Artificială***

**Anul II Licenţă, Specialităţile *Informatică, Informatică aplicată, Management Informaţional*.**

**Durata examenului: 3 ore academice**

**1.** Пусть задана следующая программа: cifra(7). cifra(8). n(A,B,C):- cifra(C),!, cifra(A), cifra(B).

Ответьте на следующие недетерминированные вопросы:

а) Goal: n(A,B,C)

Ответ:

1. A = B, B = C, C = 7

2. A = C, C = 7, B = 8

3. A = 8, B = C, C = 7

4. A = B, B = 8, C = 7

б) Goal: n(7,8,9)

Ответ:

false

в) Goal: n(7,8,C)

Ответ:

C = 7

г) Goal: n(A,B,8)

Ответ:

1. A = B, B = 7

2. A = 7, B = 8

3. A = 8, B = 7

4. A = B, B = 8

д) Goal: n(A,8,8)

Ответ:

1. A = 7

2. A = 8

(15 п.)

**2.** Пусть задан граф, определяющий функцию перехода для некоторого конечного автомата.

**s0**

**s1**

**s2**

**s3**

**s4**

**s5**

**s6**

**s7**

**a**

**b**

**a**

**b**

**c**

**d**

**d**

**d**

**e**

Начальным состоянием является s0, конечным – s7.

а) Найдите общий вид слов, распознаваемых этим

автоматом.

Ответ: cde

б) Укажите минимальное слово, распознаваемое

автоматом.

Ответ: c(abba)^n d(d)^m e, m>=2, чётн. n>0 нечётн.

в) Укажите одно слово, распознаваемое автоматом,

имеющее длину L: 15 < L < 21.

Ответ: cababcababcababde

(15 п.)

**3.** Написать Пролог-программу которая проверяет, если в символьном списке встречаются первые четыре буквы вашей фамилии, в обратном порядке. Главный предикат должен иметь один единственый аргумент – список для проверки.

Пример. Если ваша фамилия «Иванов» (первые 4 буквы: ‘i’, ‘v’, ‘a’, ‘n’), тогда:

Goal: main([а, y, n, c, c, a, d, v, n, n, i, q]) → “Yes”

Goal: main([v, а, x, x, a, d, n, n, i, a]) → “No” (20 п.)

Код программы:

f(X,Y):-select(X,Y,R),\+member(X, R).

check ([],\_).

check ([X|XZ],[X|XZZ]) :- f(X,[X|XZZ]), check ( XZ, XZZ ).

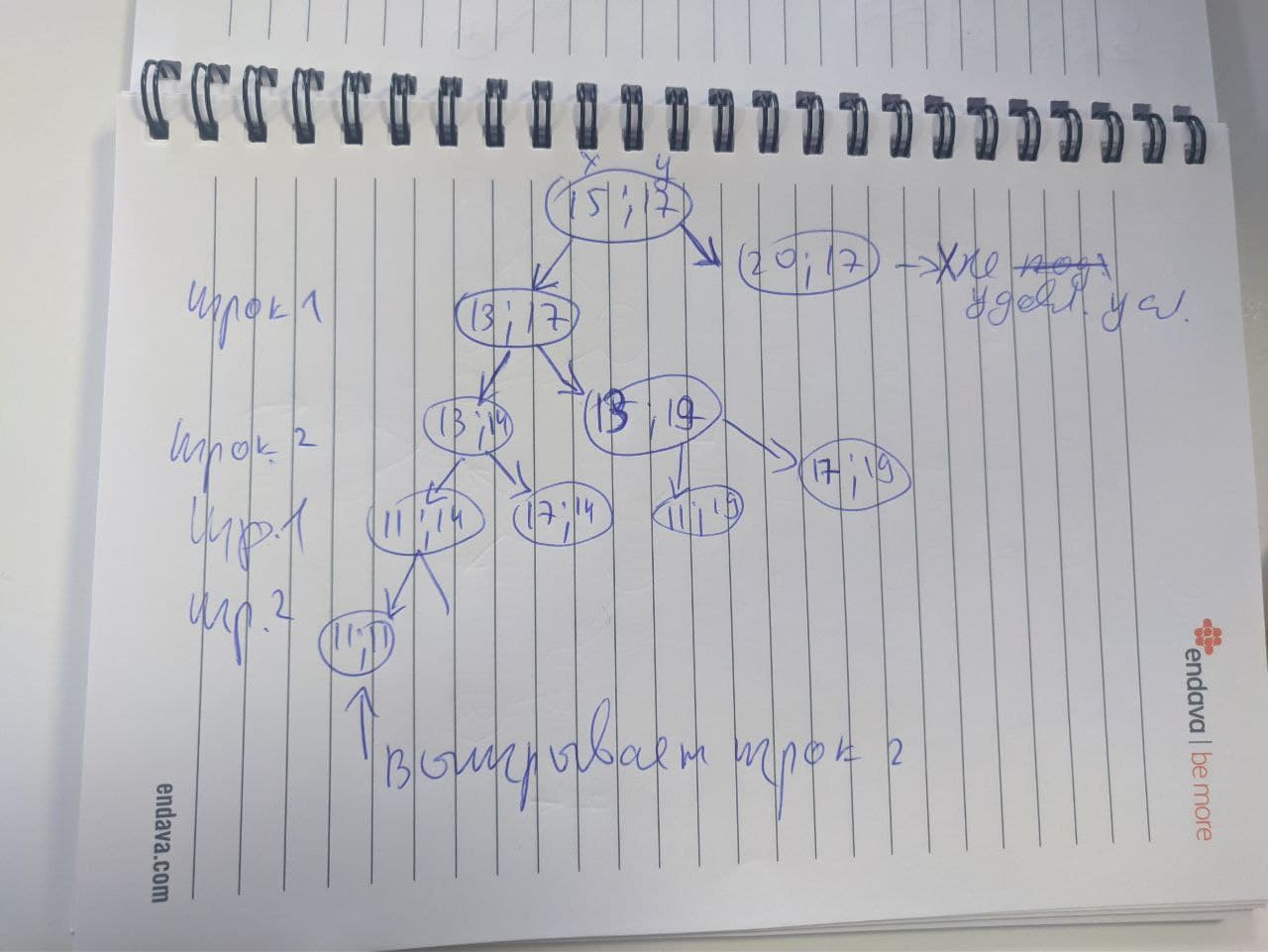
check ([X|XZ],[\_|XZZ]) :- check ( [X|XZ], XZZ ).

Goal: main([c,i,o,b],[a,c,s,i,n,m,o,w,e,b,t,y])

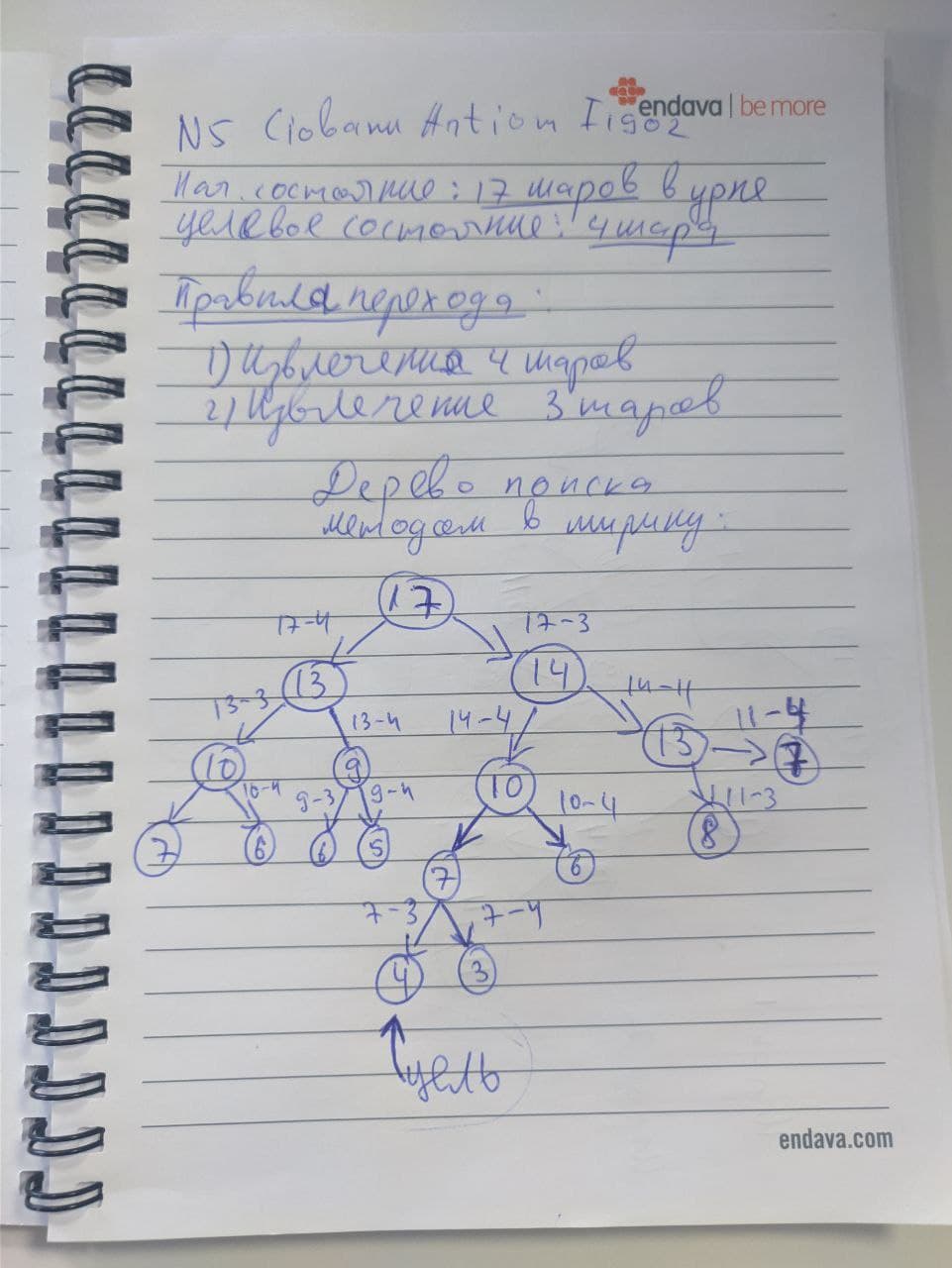
****

**4.** Рассмотрим следующую игру. Даны два числа **X**=15 и **Y**=17. Два игрока поочередно модифицируют эти числа следующим образом: первый игрок может уменьшить на 2 или увеличить на 5 число **X**, а второй игрок может уменьшить на 3 или увеличить на 2 число **Y**, но с условием что после модификации числа должны удовлетворять: 9<**X**<20 и 9<**Y**<20. Выигрывает игрок, который своим ходом уравнивает числа: **X**=**Y**. Также, считается проигрышной позиция игры **X**=14, **Y**=13.

Построить игровое дерево и указать один из основных вариантов игры. (25 п.)



**5.** Представьте следующую задачу через пространство возможных состояний: *„В некоторой урне содержатся 17 шаров. Найти последовательность действий, которая оставит в урне ровно 4 шара. Допустимые действия: а) из урны можно извлечь одновременно 4 шара, б) из урны можно извлечь одновременно 3 шара”*. Опишите состояния задачи, начальные и целевые состояния, правила перехода из одного состояния в другое. Для данной задачи нарисуйте дерево поиска, построенное методом поиска **в ширину**. Если в некоторой ветви дерева повторяется одно из состояний задачи, данная ветвь далее не исследуется. Построение дерева завершается при достижении одного из целевых состояний. (25 п.)



**BAREM DE NOTARE**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nota** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Punctaj acumulat** | **E** | **0--10** | **11--20** | **21--29** | **30--45** | **46--60** | **61--75** | **76--85** | **86--95** | **96--100** |
| **Examinator: Viorel Grigorcea, lect.univ.** | | | | | **\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ 2021 \_\_\_\_\_\_\_\_\_\_\_\_\_** | | | | | |