Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Институт/Факультет | Институт информационных технологий и анализа | | | |
|  |  | наименование | |  |
| данных | |  | |  |
| Кафедра/Структурное подразделение | | | Автоматизированные системы | |
|  | | наименование (при наличии) | | |
| обработки информации и управления | | | | |

**ОТЧЕТ**

по лабораторной работе № \_2\_\_\_

по дисциплине \_\_Системы искусственного интеллекта\_\_\_\_

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил | АСУб-20-2 |  |  |  | Арбакова А.В. |
|  | шифр группы |  | подпись |  | Фамилия И.О. |
| Проверил |  |  |  |  | Столбов А.Б. |
|  | должность |  | подпись |  | Фамилия И.О. |

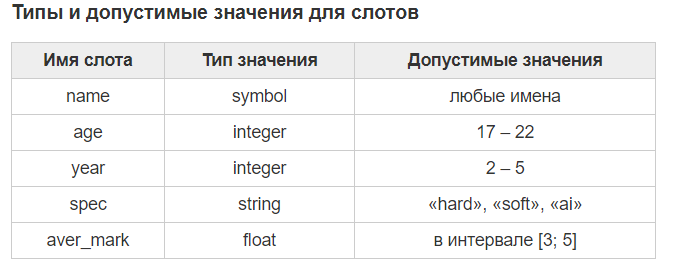
Иркутск – 20 \_23\_

**Лабораторная работа №2: Построение ЭС с использованием неупорядоченных фактов (шаблонов) и различных типов условных элементов в антецедентах правил**

**Цель работы:** Изучение примеров программирования экспертной системы с использованием неупорядоченных фактов (шаблонов) и различных типов условных элементов в антецедентах правил.

**Задание:**

Необходимо сформировать модельную базу данных с помощью команд deftemplate и deffacts. Далее используя только литеральные ограничения, составить правила для нахождения в БД фактов, удовлетворяющих заданным условиям. Применить УЭ-проверки в составленных правилах. Изменить сформированные правила путем добавления в антецеденты условных элементов с ограничением по возвращаемому значению.



**Порядок выполнения работы:**

1. Сформировать базу данных, содержащую не менее десяти неупорядоченных фактов на основе следующего шаблона:

(deftemplate student

(slot name) ; имя студента

(slot age) ; возраст

(slot year) ; год обучения (курс)

(slot spec) ; специализация

(slot aver\_mark)) ; средний балл

**Решение**:

(deftemplate student

(slot name)

(slot age)

(slot year)

(slot spec)

(slot aver\_mark))

(deffacts students

(student (name John) (age 20) (year 3) (spec "hard") (aver\_mark 4.5))

(student (name Jane) (age 18) (year 2) (spec "hard") (aver\_mark 5.0))

(student (name Jude) (age 22) (year 4) (spec "soft") (aver\_mark 3.8))

(student (name Toma) (age 21) (year 3) (spec "ai") (aver\_mark 3.5))

(student (name Joan) (age 19) (year 4) (spec "hard") (aver\_mark 4.5))

(student (name Mark) (age 23) (year 5) (spec "soft") (aver\_mark 3.0))

(student (name Fred) (age 20) (year 2) (spec "ai") (aver\_mark 4.0))

(student (name Eric) (age 24) (year 6) (spec "hard") (aver\_mark 4.5))

(student (name Mary) (age 20) (year 3) (spec "ai") (aver\_mark 3.6))

(student (name Beth) (age 24) (year 5) (spec "soft") (aver\_mark 3.1))

)

1. Сохранить конструкции deftemplate и deffacts в файле.
2. Составить в соответствии с вариантом задания правила, реализующие описанные ниже функции, с использованием заданных типов условных элементов. Правила, соответствующие различным пунктам задания, следует сохранять в разных файлах, чтобы демонстрировать их работу преподавателю по отдельности.
   1. Используя только литеральные ограничения, составить правила для нахождения в БД фактов, удовлетворяющих заданным в табл. 1 условиям, и выдачи соответствующих сообщений.

Таблица 1

| **№ варианта** | **Условие в антецеденте правила** | **Сообщение, выводимое в консеквенте правила** |
| --- | --- | --- |
| 1 | Студент 2-го курса | Студент 2-го курса <name> учится по специализации <spec>. |

**Решение**:

(deftemplate student

(slot name)

(slot age)

(slot year)

(slot spec)

(slot aver\_mark))

(deffacts students

(student (name John) (age 20) (year 3) (spec "hard") (aver\_mark 4.5))

(student (name Jane) (age 18) (year 2) (spec "hard") (aver\_mark 5.0))

(student (name Jude) (age 22) (year 4) (spec "soft") (aver\_mark 3.8))

(student (name Toma) (age 21) (year 3) (spec "ai") (aver\_mark 3.5))

(student (name Joan) (age 19) (year 4) (spec "hard") (aver\_mark 4.5))

(student (name Mark) (age 23) (year 5) (spec "soft") (aver\_mark 3.0))

(student (name Fred) (age 20) (year 2) (spec "ai") (aver\_mark 4.0))

(student (name Eric) (age 24) (year 6) (spec "hard") (aver\_mark 4.5))

(student (name Mary) (age 20) (year 3) (spec "ai") (aver\_mark 3.6))

(student (name Beth) (age 24) (year 5) (spec "soft") (aver\_mark 3.1))

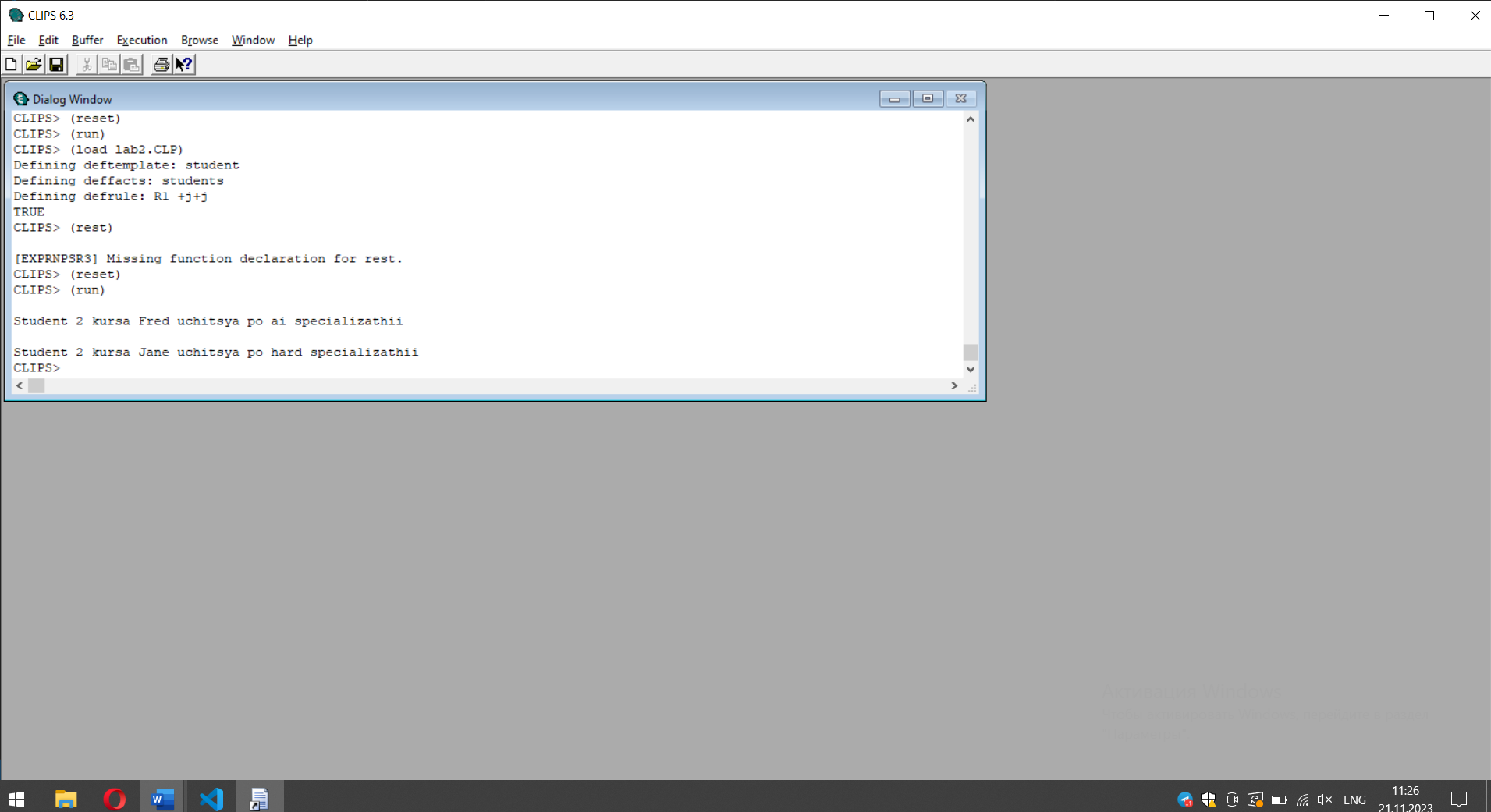
)

(defrule R1

(student (name ?name) (age ?age) (year 2) (spec ?spec) (aver\_mark ?aver\_mark))

=>

(printout t crlf "Student 2 kursa " ?name " uchitsya po " ?spec " specializathii" crlf ))



* 1. Изменить сформированные в п. a. правила путем добавления в антецедент новых условий и изменения выводимых сообщений в соответствии с табл. 2. При реализации новых УЭ использовать УЭ-проверки (test-CE).

| **№ варианта** | **Условие в антецеденте правила** | **Сообщение, выводимое в консеквенте правила** |
| --- | --- | --- |
| 1 | Студент 2-го курса, средний балл не ниже 4.5 | Студент <name> имеет средний балл <aver\_mark>. |

**Решение**:

(deftemplate student

(slot name)

(slot age)

(slot year)

(slot spec)

(slot aver\_mark))

(deffacts students

(student (name John) (age 20) (year 3) (spec "hard") (aver\_mark 4.5))

(student (name Jane) (age 18) (year 2) (spec "hard") (aver\_mark 5.0))

(student (name Jude) (age 22) (year 4) (spec "soft") (aver\_mark 3.8))

(student (name Toma) (age 21) (year 3) (spec "ai") (aver\_mark 3.5))

(student (name Joan) (age 19) (year 4) (spec "hard") (aver\_mark 4.5))

(student (name Mark) (age 23) (year 5) (spec "soft") (aver\_mark 3.0))

(student (name Fred) (age 20) (year 2) (spec "ai") (aver\_mark 4.0))

(student (name Eric) (age 24) (year 6) (spec "hard") (aver\_mark 4.5))

(student (name Mary) (age 20) (year 3) (spec "ai") (aver\_mark 3.6))

(student (name Beth) (age 24) (year 5) (spec "soft") (aver\_mark 3.1))

)

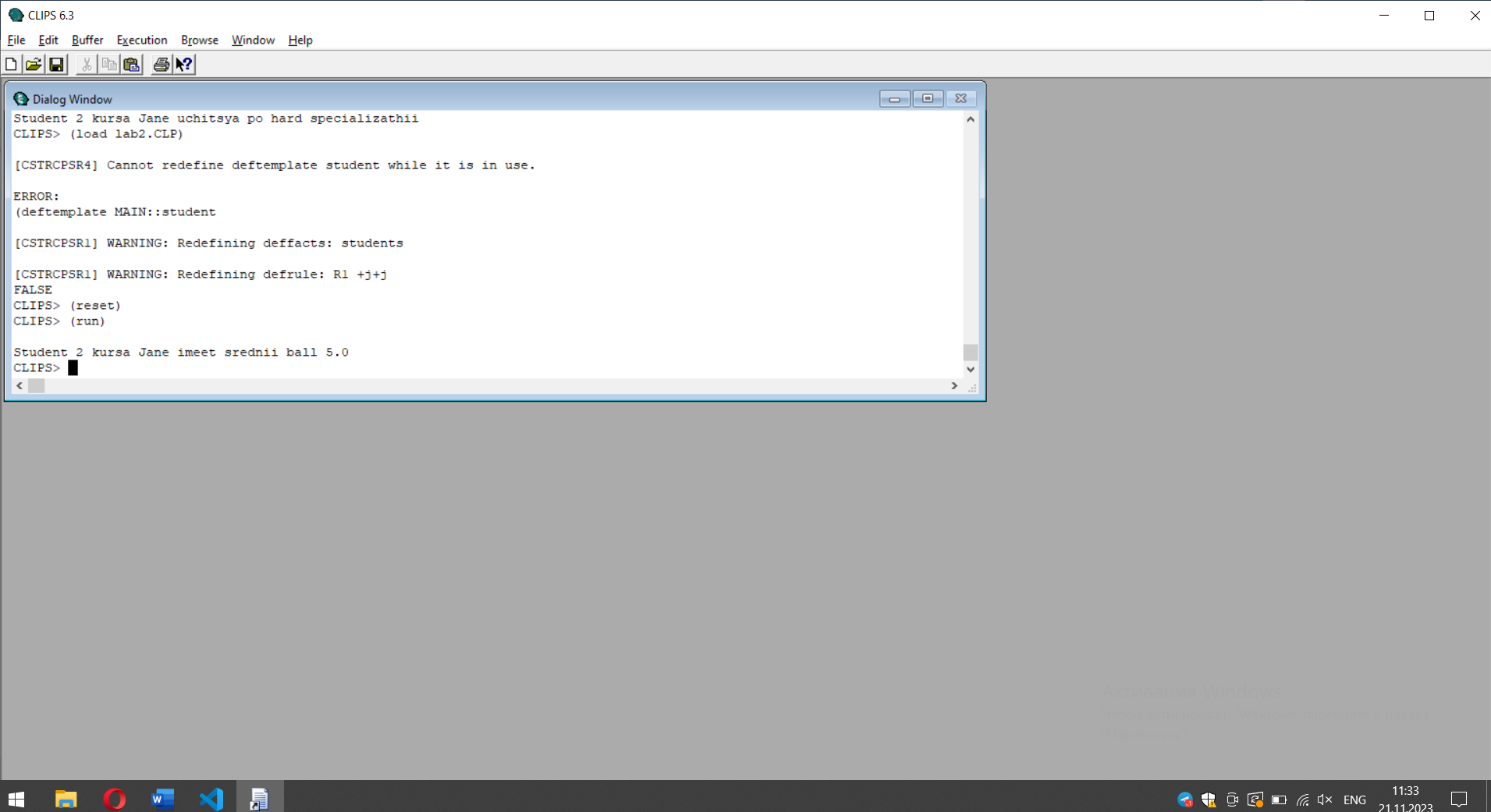
(defrule R1

(student (name ?name) (age ?age) (year ?year) (spec ?spec) (aver\_mark ?aver\_mark))

(test (and(= ?year 2) (> ?aver\_mark 4.5)))

=>

(printout t crlf "Student 2 kursa " ?name " imeet srednii ball " ?aver\_mark " " crlf ))



* 1. Изменить сформированные в п. b. правила путем добавления в антецеденты предикатных условных элементов для проверки типов значений слотов в соответствии с табл. 3.

| **№ варианта** | **Проверяемые значения** | **Сообщение, выводимое в консеквенте правила** |
| --- | --- | --- |
| 1 | <year>, <aver\_mark> | Студент <name>: тип значения в слоте <year> – integer, а в слоте <aver\_mark> – float. |

**Решение**:

(deftemplate student

(slot name)

(slot age)

(slot year)

(slot spec)

(slot aver\_mark))

(deffacts students

(student (name John) (age 20) (year 3) (spec "hard") (aver\_mark 4.5))

(student (name Jane) (age 18) (year 2) (spec "hard") (aver\_mark 5.0))

(student (name Jude) (age 22) (year 4) (spec "soft") (aver\_mark 3.8))

(student (name Toma) (age 21) (year 3) (spec "ai") (aver\_mark 3.5))

(student (name Joan) (age 19) (year 4) (spec "hard") (aver\_mark 4.5))

(student (name Mark) (age 23) (year 5) (spec "soft") (aver\_mark 3.0))

(student (name Fred) (age 20) (year 2) (spec "ai") (aver\_mark 4.0))

(student (name Eric) (age 24) (year 6) (spec "hard") (aver\_mark 4.5))

(student (name Mary) (age 20) (year 3) (spec "ai") (aver\_mark 3.6))

(student (name Beth) (age 24) (year 5) (spec "soft") (aver\_mark 3.1))

)

(defrule R1

(student (name ?name) (age ?age) (year ?year&:(numberp ?year)) (spec ?spec) (aver\_mark ?ave\_mark&:(floatp ?ave\_mark)))

(test (and(= ?year 2) (> ?aver\_mark 4.5)))

=>

(printout t crlf "Student 2 kursa " ?name " imeet srednii ball " ?aver\_mark " " crlf ))

* 1. Изменить сформированные в п. c. правила путем добавления в антецеденты условных элементов с ограничением по возвращаемому значению. Условия ограничения для различных вариантов приведены в табл. 4.

| **№ варианта** | **Условный элемент с ограничением по возвращаемому значению** | **Сообщение, выводимое в консеквенте правила** |
| --- | --- | --- |
| 1 | Поступил в университет в возрасте не старше 20 лет | Студент <name> поступил в университет в возрасте не старше 20 лет. |

**Решение**:

(deftemplate student

(slot name)

(slot age)

(slot year)

(slot spec)

(slot aver\_mark))

(deffacts students

(student (name John) (age 20) (year 3) (spec "hard") (aver\_mark 4.5))

(student (name Jane) (age 18) (year 2) (spec "hard") (aver\_mark 5.0))

(student (name Jude) (age 22) (year 4) (spec "soft") (aver\_mark 3.8))

(student (name Toma) (age 21) (year 3) (spec "ai") (aver\_mark 3.5))

(student (name Joan) (age 19) (year 4) (spec "hard") (aver\_mark 4.5))

(student (name Mark) (age 23) (year 5) (spec "soft") (aver\_mark 3.0))

(student (name Fred) (age 20) (year 2) (spec "ai") (aver\_mark 4.0))

(student (name Eric) (age 24) (year 6) (spec "hard") (aver\_mark 4.5))

(student (name Mary) (age 20) (year 3) (spec "ai") (aver\_mark 3.6))

(student (name Beth) (age 24) (year 5) (spec "soft") (aver\_mark 3.1))

)

(defrule R1

(student (name ?name) (age ?age) (year ?year) (spec ?spec) (aver\_mark ?ave\_mark))

(test (>= 20 (- ?age ?year)))

=>

(printout t crlf ?name " поступил в " ?age " лет" crlf))