Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет» (ПНИПУ)

Кафедра вычислительной математики и механики

**Лабораторная работа № 1**

**по дисциплине: «Интеллектуальные ИСИТ»**

Выполнил

студент группы ИСТ-19-2б

Кошкаров П.С.

Проверил

ассистент кафедры ВММБ

Нетбай Г.В.

Пермь, 2022

# Продукционные системы

Продукционная система – система принятия решения, основанная на следовании правилам (продукциям). Система моделирует человеческое мышление.

Правила, из которых состоит система, имеют название, условие (левую часть), и действие/следствие (правую часть).

# CLIPS

CLIPS – язык программирования, программа, позволяющая вводить правила для принятия решений. Программу с установленными правилами можно запустить для работы с вводом пользователя. Конечным результатом станет вывод, результат, в соответствии с правилами.

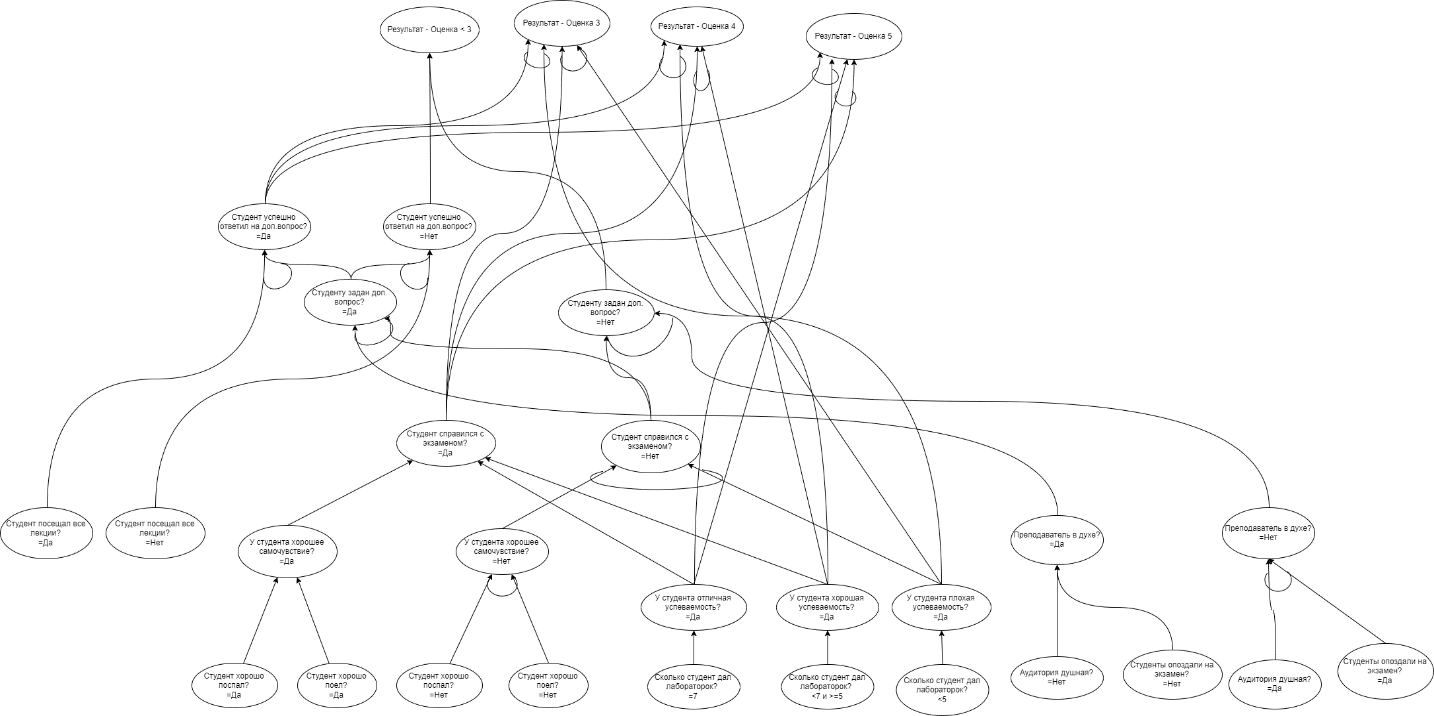
# Тема варианта

“Оценка на экзамене”

# Требования

1. Граф И/ИЛИ должен иметь высоту не менее 4, т.е., как минимум, от одного факта до заключения должно быть 4 правила.
2. Количество правил должно быть не менее 15.
3. Скрипт в CLIPS должен соответствовать требованиям:
   1. Часть фактов должно инициализироваться при запуске, остальные – в интерактивном режиме работы с пользователем.
   2. Должны использоваться функции (deffunction).
   3. Код должен быть единообразно отформатирован.
   4. Должны присутствовать комментарии.
   5. Должны присутствовать не только логические факты, но и целочисленные.

# Граф “И/ИЛИ”



# Нотация “IF-THEN”

|  |  |
| --- | --- |
| Правило:  У студента хорошее самочувствие. | **IF** “Студент хорошо поел” = “Да” **OR** “Студент хорошо поспал” = “Да” **THEN** “У студента хорошее самочувствие” = “Да” |
| Правило:  У студента плохое самочувствие. | **IF** “Студент хорошо поел” = “Нет” **AND** “Студент хорошо поспал” = “Нет” **THEN** “У студента хорошее самочувствие” = “Нет” |
| Правило:  У студента отличная успеваемость. | **IF** “Студент сдал лабороторок” = “7” **THEN** “У студента отличная успеваемость” = “Да” |
| Правило:  У студента хорошая успеваемость. | **IF** “Студент сдал лабороторок” = “<7 и >=5” **THEN** “У студента хорошая успеваемость” = “Да” |
| Правило:  У студента плохая успеваемость. | **IF** “Студент сдал лабороторок” = “<5” **THEN** “У студента плохая успеваемость” = “Да” |
| Правило:  Преподаватель в духе. | **IF** “Аудитория душная” = “Нет” **OR** “Студенты опоздали на экзамен” = “Нет” **THEN** “Преподаватель в духе” = “Да” |
| Правило:  Преподаватель не в духе. | **IF** “Аудитория душная” = “Да” **AND** “Студенты опоздали на экзамен” = “Да” **THEN** “Преподаватель в духе” = “Нет” |
| Правило:  Студент справился с экзаменом. | **IF** “У студента хорошее самочувствие” = “Да” **OR** “У студента отличная успеваемость” = “Да” **OR** “У студента хорошая успеваемость” = “Да” **THEN** “Студент справился с экзаменом” = “Да” |
| Правило:  Студент не справился с экзаменом. | **IF** “У студента хорошее самочувствие” = “Нет” **AND** “У студента плохая успеваемость” = “Да” **THEN** “Студент справился с экзаменом” = “Нет” |
| Правило:  Студенту задан доп.вопрос. | **IF** “Студент справился с экзаменом” = “Нет” **AND** “Преподаватель в духе” = “Да” **THEN** “Студенту задан доп.вопрос” = “Да” |
| Правило:  Студенту не задан доп.вопрос. | **IF** (“Студент справился с экзаменом” = “Нет” **AND** “Преподаватель в духе” = “Нет”) **THEN** “Студенту задан доп.вопрос” = “Нет” |
| Правило:  Студент успешно ответил на доп.вопрос. | **IF** “Студенту задан доп.вопрос” = “Да” **AND** “Студент посещал все лекции” = “Да” **THEN** “Студенту успешно ответил на доп.вопрос” = “Да” |
| Правило:  Студент не успешно ответил на доп.вопрос. | **IF** “Студенту задан доп.вопрос” = “Да” **AND** “Студент посещал все лекции” = “Нет” **THEN** “Студенту успешно ответил на доп.вопрос” = “Нет” |
| Результат-правило:  Оценка == 2 | **IF** “Студенту задан доп.вопрос” = “Нет” **OR** “Студент успешно ответил на доп.вопрос” = “Нет” **THEN** “Результат”= “Оценка == 2” |
| Результат-правило:  Оценка == 3 | **IF** (“Студент успешно ответил на доп.вопрос” = “Да” **AND** “У студента плохая успеваемость” = “Да”) **OR** (“Студент справился с экзаменом” = “Да” **AND** “У студента плохая успеваемость” = “Да”) **THEN** “Результат”= “Оценка == 3” |
| Результат-правило:  Оценка == 4 | **IF** (“Студент успешно ответил на доп.вопрос” = “Да” **AND** “У студента хорошая успеваемость” = “Да”) **OR** (“Студент справился с экзаменом” = “Да” **AND** “У студента хорошая успеваемость” = “Да”) **THEN** “Результат”= “Оценка == 4” |
| Результат-правило:  Оценка == 5 | **IF** (“Студент успешно ответил на доп.вопрос” = “Да” **AND** “У студента отличная успеваемость?” = “Да”) **OR** (“Студент справился с экзаменом” = “Да” **AND** “У студента отличная успеваемость” = “Да”) **THEN** “Результат”= “Оценка == 5” |

# Листинг кода CLIPS

(deffunction ask\_value (?question)

(print ?question)

(bind ?answer (read))

?answer

)

(deffunction ask-question (?question $?allowed-values)

(print ?question)

(bind ?answer (read))

(if (lexemep ?answer)

then (bind ?answer (lowcase ?answer))

)

(while (not (member$ ?answer ?allowed-values)) do

(print ?question)

(bind ?answer (read))

(if (lexemep ?answer)

then (bind ?answer (lowcase ?answer))

)

)

?answer

)

(deffunction yes-or-no (?question)

(bind ?response (ask-question ?question yes no y n))

(if (or (eq ?response yes) (eq ?response y))

then yes

else no

)

)

(defrule determenite\_student\_sleep

(not (solution\_mark ?))

(not (student\_in\_mood ?))

(not (student\_good\_sleep ?))

=>

(assert (student\_good\_sleep (yes-or-no "Student have good sleep?: ")))

)

(defrule determenite\_student\_eat

(not (solution\_mark ?))

(not (student\_in\_mood ?))

(not (student\_good\_eat ?))

=>

(assert (student\_good\_eat (yes-or-no "Student have good breakfast?: ")))

)

(defrule determenite\_student\_perfomance

(not (solution\_mark ?))

(not (student\_perfomance ?))

=>

(assert (student\_perfomance (ask\_value "How muck homeworks student reach?: ")))

)

(defrule determenite\_lections

(not (solution\_mark ?))

(not (student\_was\_on\_lections ?))

=>

(assert (student\_was\_on\_lections (yes-or-no "Student went on all lections?: ")))

)

(defrule determenite\_auditory

(not (solution\_mark ?))

(not (teacher\_in\_mood ?))

(not (auditory\_good ?))

=>

(assert (auditory\_good (yes-or-no "Auditory good?: ")))

)

(defrule determenite\_students

(not (solution\_mark ?))

(not (teacher\_in\_mood ?))

(not (students\_fast ?))

=>

(assert (students\_fast (yes-or-no "Students were before begin exam?: ")))

)

(defrule determenite\_student\_mood\_rule1

(or

(student\_good\_sleep yes)

(student\_good\_eat yes)

)

=>

(assert (student\_in\_mood yes))

(print "Student in mood: Yes" crlf)

)

(defrule determenite\_student\_mood\_rule2

(and

(student\_good\_sleep no)

(student\_good\_eat no)

)

=>

(assert (student\_in\_mood no))

(print "Student in mood: No" crlf)

)

(defrule determenite\_student\_end\_perfomance\_best

(student\_perfomance ?value)

(not (student\_best\_perfomance ?))

(not (student\_good\_perfomance ?))

(not (student\_bad\_perfomance ?))

(test (eq ?value 7))

=>

(assert (student\_best\_perfomance yes))

(print "Student have perfoming: Best" crlf)

)

(defrule determenite\_student\_end\_perfomance\_good

(student\_perfomance ?value)

(not (student\_best\_perfomance ?))

(not (student\_good\_perfomance ?))

(not (student\_bad\_perfomance ?))

(test (and (< ?value 7) (>= ?value 5)))

=>

(assert (student\_good\_perfomance yes))

(print "Student have perfoming: Good" crlf)

)

(defrule determenite\_student\_end\_perfomance\_bad

(student\_perfomance ?value)

(not (student\_best\_perfomance ?))

(not (student\_good\_perfomance ?))

(not (student\_bad\_perfomance ?))

(test (< ?value 5))

=>

(assert (student\_bad\_perfomance yes))

(print "Student have perfoming: Bad" crlf)

)

(defrule determenite\_teacher\_mood\_rule1

(or

(auditory\_good no)

(students\_fast no)

)

=>

(assert (teacher\_in\_mood no))

(print "Teacher in mood: No" crlf)

)

(defrule determenite\_teacher\_mood\_rule2

(and

(auditory\_good yes)

(students\_fast yes)

)

=>

(assert (teacher\_in\_mood yes))

(print "teacher\_in\_mood: Yes" crlf)

)

(defrule determenite\_student\_exam\_good

(or

(student\_in\_mood yes)

(student\_best\_perfomance yes)

(student\_good\_perfomance yes)

)

=>

(assert (student\_exam\_good yes))

(print "Student exam is good: Yes" crlf)

)

(defrule determenite\_student\_exam\_bad

(and

(student\_in\_mood no)

(student\_bad\_perfomance yes)

)

=>

(assert (student\_exam\_good no))

(print "Student exam is good: No" crlf)

)

(defrule determenite\_student\_add\_question\_yes

(and

(student\_exam\_good no)

(teacher\_in\_mood yes)

)

=>

(assert (student\_add\_question yes))

(print "Student have add question: Yes" crlf)

)

(defrule determenite\_student\_add\_question\_no

(and

(student\_exam\_good no)

(teacher\_in\_mood no)

)

=>

(assert (student\_add\_question no))

(print "Student have add question: No" crlf)

)

(defrule determenite\_student\_add\_question\_good

(and

(student\_was\_on\_lections yes)

(student\_add\_question yes)

)

=>

(assert (student\_add\_question\_good yes))

(print "Student have add question is good: Yes" crlf)

)

(defrule determenite\_student\_add\_question\_bad

(and

(student\_was\_on\_lections no)

(student\_add\_question yes)

)

=>

(assert (student\_add\_question\_good no))

(print "Student have add question is bad: No" crlf)

)

(defrule determenite\_student\_mark\_2

(or

(student\_add\_question no)

(student\_add\_question\_good no)

)

(not (solution\_mark ?))

=>

(assert (solution\_mark "Mark 2"))

(print "Mark 2" crlf)

)

(defrule determenite\_student\_mark\_3

(or

(and

(student\_bad\_perfomance yes)

(student\_add\_question\_good yes)

)

(and

(student\_exam\_good yes)

(student\_bad\_perfomance yes)

)

)

(not (solution\_mark ?))

=>

(assert (solution\_mark "Mark 3"))

(print "Mark 3" crlf)

)

(defrule determenite\_student\_mark\_4

(or

(and

(student\_good\_perfomance yes)

(student\_add\_question\_good yes)

)

(and

(student\_exam\_good yes)

(student\_good\_perfomance yes)

)

)

(not (solution\_mark ?))

=>

(assert (solution\_mark "Mark 4"))

(print "Mark 4" crlf)

)

(defrule determenite\_student\_mark\_5

(or

(and

(student\_best\_perfomance yes)

(student\_add\_question\_good yes)

)

(and

(student\_exam\_good yes)

(student\_best\_perfomance yes)

)

)

(not (solution\_mark ?))

=>

(assert (solution\_mark "Mark 5"))

(print "Mark 5" crlf)

))