Министерство образования Республики Беларусь

Учреждение образования

“Брестский государственный технический университет”

Кафедра интеллектуально-информационных технологий

Лабораторная работа №6

“Разработка логической базы знаний для автоматизации

указанного вида деятельности”

Выполнил:

студент 3 курса

группы ИИ-21

Романко Н.А.

Проверил:

Савонюк В.А.

Брест 2024

**Цель работы:**

1. Разработать логическую базу знаний для автоматического управляющего устройства.
2. Получить навыки анализа предметной области и разработки логических баз знаний.

**Задание:**

5. Ваша программа должна описывать базу знаний для для автоматического постоянного управления освещением витрины магазина.

Освещение витрины управляется по трем группам. Датчик движения и

выключатель сумерек обеспечивают правильное освещение витрины.

Три группы освещения управляются по-разному:

* Первой группой освещения управляет выключатель времени.
* Вторая группа освещения включается дополнительно, когда становится темнее.
* Необходимо, чтобы третья группа освещения включилась, когда темно, когда датчик движения был активирован и выключатель времени не включен.

Если человек приближается к витрине, датчик движения заставит неоновый знак включиться в течение определенного периода.

Главный выключатель позволяет вам включать и включать систему.

Кнопка «Проверка ламп» используется, чтобы включить все освещение в течение одной минуты.

**Решение**

В первую очередь определим соответствия входных и выходных сигналов проектируемой системы переменным. Условимся обозначать все входные сигналы переменными I с соответствующим индексом, а все выходные сигналы - переменными Q с соответствующим индексом.

**Входные переменные:**

I1 – сигнал датчика времени

I2 – сигнал датчика света

I3 - сигнал датчика движения

I4 – сигнал проверки ламп

**Выходные переменные:**

Q1 – включение\выключение первой группы освещения

Q2 – включение\выключение второй группы освещения

Q3 – включение\выключение третьей группы освещения

Q4 – Индикатор состояния главного перекюлчателя(включен\выключен)

Далее разобьем нашу задачу на условные части и составим логические высказывания для каждой из частей.

**Управление датчиком времени:**

Условие: Если получен сигнал датчика времени(I1).

Логическое высказывание: I1 == True.

**Управление датчиком света:**

Условие: Если получен сигнал датчика света (I2).

Логическое высказывание: I2 == True.

**Управление датчиком движения:**

Условие: Если получен сигнал от датчика движения(I3).

Логическое высказывание: I3 == True.

**Система выключена:**

Условие: Если главный переключатель выключен (Q4 == False).

Логическое высказывание: Q4 == False.

**Система включена:**

Условие: Если главный переключатель включен (Q4 == True).

Логическое высказывание: Q4 == True.

**Проверка ламп:**

Условие: Если получен сигнал проверки ламп (I4 == True).

Логическое высказывание: I4 == True.

Эта база знаний описывает систему управления освещением витрины магазина с использованием предоставленных переменных и логических условий.

Далее стоит запрограммировать все это.

Код программы:

import time

class LightingControlSystem:

def \_\_init\_\_(self):

self.motion\_sensor = False

self.twilight\_switch = False

self.time\_switch = False

self.main\_switch = False

self.neon\_sign = False

def check\_lamps(self):

if not self.main\_switch:

print("Главный выключатель выключен. Включите главный выключатель перед проверкой ламп.")

return

print("Проверка ламп...")

self.turn\_on\_neon\_sign()

self.turn\_on\_first\_group\_lights()

self.turn\_on\_second\_group\_lights()

self.turn\_on\_third\_group\_lights()

def turn\_on\_neon\_sign(self):

if self.neon\_sign:

print("Неоновая подсветка уже включена")

else:

print("Неоновая подсветка включена")

self.neon\_sign = True

def turn\_off\_neon\_sign(self):

if not self.neon\_sign:

print("Неоновая подсветка уже выключена")

else:

print("Неоновая подсветка выключена")

self.neon\_sign = False

def turn\_on\_first\_group\_lights(self):

if self.time\_switch:

print("Первая группа освещения включена по времени")

else:

print("Первая группа освещения выключена")

def turn\_on\_second\_group\_lights(self):

if self.twilight\_switch:

print("Вторая группа освещения включена по датчикам света")

else:

print("Вторая группа освещения выключена")

def turn\_on\_third\_group\_lights(self):

if self.motion\_sensor and not self.time\_switch:

print("Третья группа освещения включена по датчику движения")

else:

print("Третья группа освещения выключена")

def toggle\_motion\_sensor(self):

if not self.main\_switch:

print("Главный выключатель выключен. Включите главный выключатель перед активацией датчика движения.")

return

self.motion\_sensor = not self.motion\_sensor

if self.motion\_sensor:

print("Датчик движения активирован")

else:

print("Датчик движения деактивирован")

def switch\_twilight(self):

if not self.main\_switch:

print("Главный выключатель выключен. Включите главный выключатель перед переключением режима ночного света.")

return

print("Ночный режим переключен")

self.twilight\_switch = not self.twilight\_switch

def switch\_time(self):

if not self.main\_switch:

print("Главный выключатель выключен. Включите главный выключатель перед переключением режима времени.")

return

print("Режим времени переключен")

self.time\_switch = not self.time\_switch

def switch\_main(self):

print("Главный выключатель переключен")

self.main\_switch = not self.main\_switch

if not self.main\_switch:

self.turn\_off\_neon\_sign()

def turn\_on\_all\_lights\_for\_x\_seconds(self, seconds):

if not self.main\_switch:

print("Главный выключатель выключен. Включите главный выключатель перед включением всех ламп.")

return

print(f"Все лампы включены на {seconds} секунд")

print("Неоновая подсветка включена")

print("Первая группа освещения включена")

print("Вторая группа освещения включена")

print("Третья группа освещения включена")

time.sleep(seconds)

def user\_input(control\_system):

while True:

print("\nВыберите действие:")

print("1. Проверить лампы")

print("2. Переключить основной светильник")

print("3. Переключить режим времени")

print("4. Переключить режим ночного света")

print("5. Включить/выключить датчик движения")

print("6. Включить все лампы на X секунд")

print("7. Выход")

choice = input("Введите ваш выбор: ")

if choice == "1":

control\_system.check\_lamps()

elif choice == "2":

control\_system.switch\_main()

elif choice == "3":

control\_system.switch\_time()

elif choice == "4":

control\_system.switch\_twilight()

elif choice == "5":

control\_system.toggle\_motion\_sensor()

elif choice == "6":

seconds = int(input("Введите количество секунд для включения всех ламп: "))

control\_system.turn\_on\_all\_lights\_for\_x\_seconds(seconds)

elif choice == "7":

print("Выход из программы...")

break

else:

print("Неверный выбор. Пожалуйста, введите верное значение.")

if \_\_name\_\_ == "\_\_main\_\_":

control\_system = LightingControlSystem()

user\_input(control\_system)

**Вывод:** научился разрабатывать логическую базу знаний для автоматизации и управления определенного вида деятельности