

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Факультет информационных технологий  
Кафедра Общей информатики

Направление подготовки 09.03.01 Информатика и вычислительная техника  
Направленность (профиль): Программная инженерия и компьютерные науки

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА**

**Удод Светланы Олеговны**

Тема работы:

**РАЗРАБОТКА ПРОГРАММЫ ДЛЯ ТЕСТИРОВАНИЯ ИНДИВИДУАЛЬНЫХ  
СПОСОБНОСТЕЙ К ПРОИЗВОЛЬНОМУ КОНТРОЛЮ ПОВЕДЕНИЯ ДЛЯ  
ПРОФЕССИОНАЛЬНОГО ОТБОРА ПЕРСОНАЛА**

**«К защите допущена»**  
заведующий кафедрой,  
д.ф.-м.н., доцент  
Пальчунов Д.Е. / .....  
(ФИО) / (подпись)  
«31» мая 2021 г.

**Руководитель ВКР**  
профессор КОИ ФИТ,  
д.ф.н., к.б.н., доцент  
Савостьянов А.Н. / .....  
(ФИО) / (подпись)  
«31» мая 2021 г.

Новосибирск, 2021

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)  
Факультет информационных технологий

Кафедра Общей информатики

(название кафедры)

Направление подготовки 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Программная инженерия и компьютерные науки

УТВЕРЖДАЮ

Зав. кафедрой Пальчунов Д.Е.

(фамилия, И., О.)

.....

(подпись)

«10» ноября 2020 г.

**ЗАДАНИЕ**

**НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ БАКАЛАВРА**

Студентке Удод Светлане Олеговне, группы 17207

(фамилия, имя, отчество, номер группы)

Тема «Разработка программы для тестирования индивидуальных способностей к произвольному контролю поведения для профессионального отбора персонала»

(полное название темы выпускной квалификационной работы)

утверждена распоряжением проректора по учебной работе от 10.11.2020 г. № 319

Срок сдачи студентом готовой работы «20» мая 2021 г.

Исходные данные (или цель работы):

Разработать программу для тестирования индивидуальных способностей к произвольному контролю поведения для профессионального отбора персонала

Структурные части работы:

Описание предметной области, описание требований к программе, реализация программной системы.

Руководитель ВКР

профессор КОИ ФИТ,

д.ф.н., к.б.н., доцент

Савостьянов А.Н. /.....

(ФИО) / (подпись)

«10» ноября 2020 г.

Задание приняла к исполнению

Удод С.О. /.....

(ФИО студента) / (подпись)

«10» ноября 2020 г.

## СОДЕРЖАНИЕ

|  |    |
|--|----|
| Определения, обозначения и сокращения .....                | 4  |
| Введение .....   | 5  |
| 1 Введение в предметную область .....                      | 7  |
| 1.1 Описание парадигмы «Стоп-сигнал» .....                 | 7  |
| 1.2 Методы вычисления времени реакции на стоп-сигнал ..... | 8  |
| 2 Требования к программе .....                             | 10 |
| 2.1 Игра «Стоп-сигнал» .....                               | 10 |
| 2.2 Описание уровней .....                                 | 11 |
| 2.3 Функциональные требования .....                        | 14 |
| 3 Реализация .....   | 16 |
| 3.1 Обзор Inquisit .....                                   | 16 |
| 3.2 Описание реализации .....                              | 16 |
| Заключение .....   | 23 |
| Список использованных источников и литературы .....        | 24 |
| Приложение А .....   | 26 |
| Приложение Б .....   | 34 |

## **ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

Контроль поведения — это психологический уровень регуляции поведения, реализующий индивидуальные ресурсы психической организации человека, обеспечивающий соотношение внутренних возможностей и внешних целей. На данный момент выделяют три составляющие контроля поведения: когнитивный контроль, эмоциональная регуляция и произвольный/волевой контроль.

Когнитивный контроль — это особенности интеллектуальных, когнитивных способностей, обеспечивающих анализ и упорядочивание внешней и внутренней среды, создание ментальных моделей ситуации и событий и оперирование ими. Когнитивный контроль позволяет предвосхищать события, планировать решение и моделировать исполнение, производить сравнение предполагаемого и реального результата.

Эмоциональный контроль включает контроль выражения эмоций, контроль эмоционального состояния, способы регуляции эмоциональных состояний, распознавание, интерпретацию и прогнозирование собственных эмоций и эмоций другого человека.

Произвольный контроль – самостоятельная регуляция эмоциональных реакций и поведения. Он отвечает за способность к произвольному торможению, за умение гибко подстраиваться под ситуационные требования в свободной и преднамеренной манере, способность произвольно перемещать, фокусировать и поддерживать внимание, насколько это необходимо.

## **ВВЕДЕНИЕ**

При приеме на работу важно учесть способности человека к произвольному контролю поведения. Ведь существуют сложные профессии, требующие большой ответственности: водитель поезда, пилот пассажирского самолета и т.д. На такую работу необходимо подобрать человека, у которого сильно развита способность произвольно управлять вниманием, направленным на объекты внешнего мира, реализовывать сложные действия в соответствии с мотивациями, а также регулировать свое поведение. К сожалению, стандартные вопросы на собеседованиях этого сделать не могут. Они лишь проверяют квалификацию работника и его умение коммуницировать. И это действительно большая проблема.

Для решения данной проблемы было предложено разработать программу на основе парадигмы «Стоп-сигнал» для тестирования индивидуальных способностей человека к произвольному контролю поведения. На данный момент подобных программ для тестирования персонала на основе данной парадигмы не обнаружено.

При помощи данной программной системы работодатель сможет оценивать сотрудников на то, удовлетворяют ли их способности определенным требованиям, и на основе полученных характеристик сможет подобрать подчиненному подходящую работу.

Целью данной работы является разработка программы для тестирования индивидуальных способностей человека к произвольному контролю поведения для профессионального отбора персонала.

Для достижения данной цели были поставлены следующие задачи:

- 1) Изучение предметной области;
- 2) Сбор требований к программной системе;
- 3) Проектирование программной системы;
- 4) Реализация программной системы для профессионального отбора персонала.

Работа представлена в 3 главах. В первой главе приводится общее описание предметной области и обзор литературы. Во второй – требования к программе. В третьей – описание используемых технологий и программной реализации.

## 1 Введение в предметную область

### 1.1 Описание парадигмы «Стоп-сигнал»

Торможение реакции – отличительный признак произвольного контроля. Эта концепция относится к подавлению ненужных или несоответствующих действий, что поддерживает гибкое и целенаправленное поведение в постоянно меняющейся среде. Мы сосредоточимся на парадигме стоп-сигнала, которая оказалась полезным инструментом для изучения торможения реакции в когнитивной психологии, когнитивной нейробиологии и психопатологии [1,2].

Парадигма «Стоп-сигнал» – это нейрокогнитивная задача, предназначенная для точного измерения времени, затрачиваемого мозгом на подавление несоответствующих двигательных реакций. Парадигма стоп-сигнала была первоначально разработана Гордоном Логаном в 1980-х годах [3].

Данная парадигма представляет собой задание, состоящее из двух типов испытаний (рис. 1).

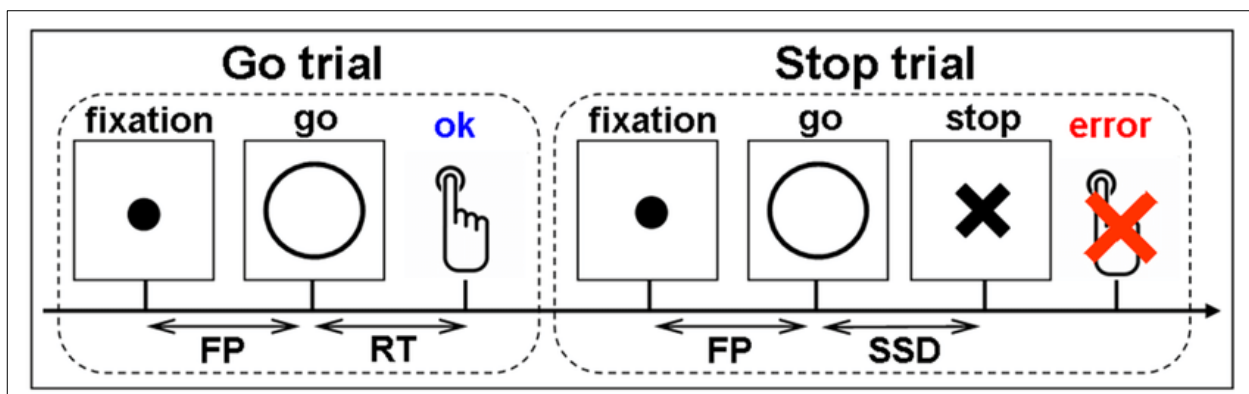


Рисунок 1 – Go и Stop испытания в задаче «Стоп-сигнал»

Первый вид испытаний это Go испытания. Для начала взгляд испытуемого фиксируется на экране и спустя некоторое время (FP – fore-period) на экране появляется целевой стимул, побуждая испытуемого быстро нажать кнопку. После нажатия кнопки или по истечении времени демонстрация стимула прекращается. Время между появлением целевого стимула и нажатием кнопки называется временем реакции (RT – reaction time) .

Второй вид испытаний – это Stop испытания. Все происходит точно также как и в Go испытании, но спустя некоторое время (SSD – stop-signal

delay) помимо целевого стимула появляется сигнал остановки. При появлении сигнала остановки испытуемый должен одернуть руку и не нажимать на кнопку.

Нажатие на любую кнопку в Stop испытании будет считаться ошибкой. В данной задаче ошибкой также будет считаться неправильно выбранная кнопка или отсутствие нажатия кнопки во время демонстрации целевого стимула без стоп сигнала.

## **1.2 Методы вычисления времени реакции на стоп-сигнал**

Как уже говорилось ранее парадигма «Стоп-сигнал» предназначена для точного измерения времени, затрачиваемого мозгом на подавление несоответствующих двигательных реакций. Это время в рамках задачи называется временем реакции на стоп-сигнал (SSRT – stop-signal reaction time). Существует несколько способов его вычисления в рамках данной парадигмы, но для начала нужно понять принцип, по которому оценивается SSRT.

SSRT оценивается в соответствии с моделью независимых гонок [3-5]: мозговая деятельность в задаче «Стоп-сигнал» может быть смоделирована как гонка между процессом Go («нажми на кнопку»), который запускается демонстрацией целевого стимула, и процессом Stop («остановись»), который запускается предъявлением сигнала остановки. Соответственно длительность процесса Go равна времени реакции на целевой стимул (RT), а длительность процесса Stop – время реакции на стоп-сигнал (SSRT).

Если процесс Go завершается до завершения процесса Stop (т. е. когда  $RT < (SSRT + SSD)$ ), то запрещение ответа на целевой стимул прошло неуспешно и выполняется ответ, то есть испытуемый всё-таки нажимает на кнопку. А если процесс Stop завершается до завершения Go процесса (т. е. когда  $RT > (SSRT + SSD)$ ), то ответ на целевой стимул удерживается, и пользователь не нажимает на кнопку.

Существует два общих метода для оценки SSRT [4]: метод интеграции и метод среднего. Метод среднего предполагает, что верна формула (1).



$$\overline{RT} = SSRT + \overline{SSD} \quad (1)$$

Поэтому SSRT можно легко оценить, вычитая среднее значение SSD из среднего RT [6].

Среднее SSD – это время задержки, при котором испытуемый подавляет свои ответы в 50% случаев (  $p(\text{ответ} \mid \text{стоп-сигнал}) = 0,50$  ).

Метод интегрирования предполагает, что время завершения процесса остановки соответствует n-му RT, где n равно количеству RT в распределении RT, умноженному на общее  $p(\text{ответ} \mid \text{стоп-сигнал})$  [7]. Затем SSRT можно оценить по формуле (2) путем вычитания среднего SSD из n- го RT [8,9].

$$SSRT = nRT - \overline{SSD} \quad (2)$$

Найденное время, необходимое для остановки реакции, широко используется в качестве клинического показателя тормозящего контроля, в первую очередь при исследовании синдрома дефицита внимания и гиперактивности, где у импульсивных субъектов SSRT больше [3], так как они медленнее реагируют на сигнал "стоп" и им труднее остановить уже начатое движение.

## 2 Требования к программе

Программа для тестирования индивидуальных способностей к произвольному контролю поведения должна быть основана на парадигме «Стоп-сигнал» в виде компьютерной игры, в ходе которой участники должны имитировать кормление животных.

Игра должна состоять из 11 уровней. Каждый уровень как-либо дополняет или изменяет основную игру «Стоп-сигнал»: добавлением нового стимула, уменьшением времени демонстрации стимула, изменением положения стоп-сигнала или фоновым шумом. К каждому уровню должна прилагаться инструкция с новыми правилами. Данные уровни позволяют посмотреть на то, как результаты прохождения одной и той же игры меняются в зависимости от условий.

### 2.1 Игра «Стоп-сигнал»

На экране монитора испытуемым в случайном порядке предъявляются два визуальных стимула («тигр» или «заяц»), которых необходимо покормить. Зайца морковкой (кнопка A), а тигра мясом (кнопка D). Испытуемому нужно либо нажать на нужную кнопку после появления целевого стимула, либо подавить свое уже подготовленное движение, при появлении визуального сигнала «стоп». Игра имеет следующие параметры:

- $N = 58$  – кол-во испытаний;
- $FP = 3-5с$  – время между целевыми стимулами;
- $T = 750мс$  – время демонстрации целевого стимула;
- $SS = 25\%$  – доля испытаний со стоп-сигналом;
- $SSD = \{0.25*RT; 0.5*RT; RT\}$  – задержка сигнала остановки;
- $p(SSD=0.5RT) = 0.5$  – вероятность того, что задержка равна половине от времени реакции;
- $p(SSD=0.25RT) = 0.25$ ;
- $p(SSD=RT) = 0.25$ ;
- $StopPos$  = в центре – положение стоп-сигнала.

На протяжении всей игры ведется счёт очков. За каждый правильный ответ пользователь получает 1 балл, а за ошибку балл отнимается.

Ошибкой в игре будет являться неправильно выбранная кнопка, нажатие нужной кнопки после появления целевого сигнала (картинки) или нажатие любой кнопки во время испытания со стоп-сигналом. Все три варианта засчитываются как ошибочные.

## **2.2 Описание уровней**

Тестовая игра. Эта игра предназначена для ознакомления пользователя с игрой и ее правилами, а также для первого вычисления средней реакции пользователя. Вычисление средней реакции должно происходить после первых 30-ти испытаний. Соответственно испытания со стоп-сигналом должны появляться только после 30-ти испытаний, иначе невозможно будет вычислить время задержки стоп-сигнала. В оставшихся 28 испытаниях процент испытаний со стоп-сигналом должен быть 40 ( $SS = 40\%$ ).

Игра с двумя целевыми стимулами. По параметрам данная игра не отличается от основной игры. Стимулами также являются заяц и тигр. Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с двумя целевыми стимулами. Если испытуемым было набрано меньше 25% ошибок, то испытуемый переходит к игре с тремя целевыми стимулами, иначе игры с большим количеством стимулов пропускаются и переход осуществляется к игре с изменением положения стоп-сигнала.

Игра с тремя целевыми стимулами. К существующим целевым стимулам, заяц и тигр, добавляется еще один – птица. Её необходимо кормить семенами (кнопка W). Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с тремя целевыми стимулами. Если испытуемым было набрано меньше 25% ошибок, то испытуемый переходит к игре с четырьмя целевыми стимулами, иначе игры с большим количеством стимулов пропускаются и переход осуществляется к игре с изменением положения стоп-сигнала.

Игра с четырьмя целевыми стимулами. К существующим целевым стимулам, заяц, тигр и птица, добавляется еще один – пингвин. Его необходимо кормить рыбой (кнопка S). Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с четырьмя целевыми стимулами. Если испытуемым было набрано меньше 25% ошибок, то испытуемый переходит к игре с пятью целевыми стимулами, иначе игра с пятью целевыми стимулами пропускается и переход осуществляется к игре с изменением положения стоп-сигнала.

Игра с пятью целевыми стимулами. К существующим целевым стимулам, добавляется еще один – мышь. Её необходимо кормить сыром (кнопка Enter). Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с пятью целевыми стимулами.

Игра с изменением положения стоп-сигнала. Положение стоп-сигнала может меняться в течение данной игры, задачей пользователя является сделать как можно меньше ошибок. Стоп-сигнал может принимать следующие положения: сверху справа, сверху посередине, сверху слева, посередине справа, в центре, посередине слева, снизу справа, снизу посередине, снизу слева.

Игра с временем демонстрации стимула 750мс. По параметрам данная игра не отличается от основной игры. Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с игрой при времени демонстрации стимула 750мс. Если испытуемым было набрано меньше 25% ошибок, то испытуемый переходит к игре с временем демонстрации стимула 650мс, иначе игры с меньшим временем демонстрации пропускаются и переход осуществляется к игре с фоновым шумом.

Игра с временем демонстрации стимула 650мс. Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с игрой при времени демонстрации стимула 650мс. Если испытуемым было набрано меньше 25% ошибок, то испытуемый переходит к игре с временем демонстрации стимула 550мс, иначе игры с меньшим временем демонстрации пропускаются и переход осуществляется к игре с фоновым шумом.

Игра с временем демонстрации стимула 550мс. Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с игрой при времени демонстрации стимула 550мс. Если испытуемым было набрано меньше 25% ошибок, то испытуемый переходит к игре с временем демонстрации стимула 450мс, иначе переход осуществляется к игре с фоновым шумом.

Игра с временем демонстрации стимула 450мс. Эта игра предназначена для того, чтобы изучить, насколько хорошо испытуемый справляется с игрой при времени демонстрации стимула 450мс.

Игра с фоновым шумом. По параметрам данная игра не отличается от основной игры. Во время прохождения данного уровня должна звучать запись шума мотора автомобиля.

Кратко переход по уровням можно представить в виде блок-схемы (рис.2).

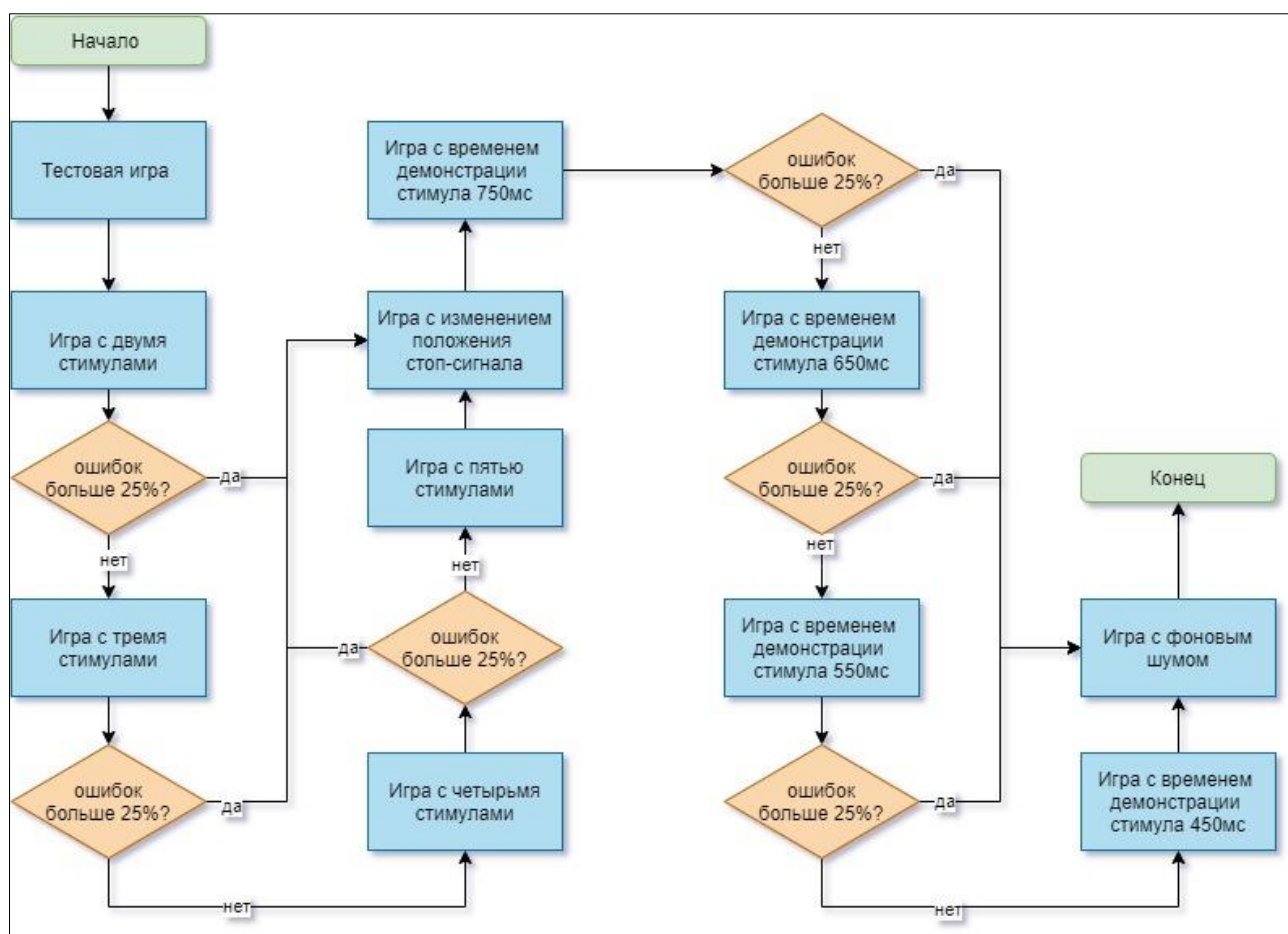


Рисунок 2 – Блок-схема перехода по уровням в игре

Все ответы пользователя на испытания должны быть сохранены в отдельный файл. Также для каждого уровня должны вычисляться следующие характеристики пользователя:

- среднее время реакции ( $\overline{RT}$ );
- процент ошибок;
- процент успешных испытаний со стоп-сигналом.

Помимо этого после прохождения всей игры должны быть вычислены следующие характеристики:

- максимальное кол-во стимулов, при которых пользователь допускает менее 25% ошибок;
- минимальное время демонстрации стимулов, при котором пользователь допускает менее 25% ошибок.

### **2.3 Функциональные требования**

На основе всего вышеперечисленного можно составить краткие функциональные требования. Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

- 1) Прохождение тестовой игры «Стоп-сигнал»;
- 2) Прохождение игры «Стоп-сигнал» на базовых настройках;
- 3) Прохождение задания «Стоп-сигнал» с тремя целевыми стимулами;
- 4) Прохождение задания «Стоп-сигнал» с четырьмя целевыми стимулами;
- 5) Прохождение задания «Стоп-сигнал» с пятью целевыми стимулами;
- 6) Прохождение задания «Стоп-сигнал» с временем демонстрации целевого стимула равным 650мс;
- 7) Прохождение задания «Стоп-сигнал» с временем демонстрации целевого стимула равным 550мс;
- 8) Прохождение задания «Стоп-сигнал» с временем демонстрации целевого стимула равным 450мс;

- 9) Прохождение задания «Стоп-сигнал» с изменением положения стоп-сигнала;
- 10) Прохождение задания «Стоп-сигнал» с фоновым шумом;
- 11) Сохранение результатов прохождения заданий.

### **3 Реализация**

#### **3.1 Обзор Inquisit**

Игра была реализована в универсальном приложении Inquisit 4, которое предназначено для разработки и проведения психологических экспериментов. Его можно использовать для реализации широкого спектра психологических задач: опросы, игры, задачи обучения и запоминания, парадигмы суждения и принятия решений и многое другое.

Эксперимент в Inquisit задается с помощью интуитивно понятного языка сценариев Inquisit. Сценарий определяет все части эксперимента: стимулы, вопросы анкеты, испытания, блоки испытаний, инструкции, логику эксперимента. Структура языка сценариев Inquisit похожа на HTML и JavaScript.

Язык сценариев Inquisit состоит из элементов и атрибутов [10]. Элементы - это основные строительные блоки сценария. Каждый элемент имеет набор атрибутов, которые определяют, как этот элемент ведет себя. Например, элемент изображения (picture), который определяет набор изображений и управляет их отображением, имеет атрибуты, которые определяют размер изображения, его расположение и т.д.

Некоторые элементы, такие как элементы expt, block и trial включают атрибуты обработчика событий, которые запускаются при запуске элемента. Они позволяют динамически получать и устанавливать свойства любого элемента во время выполнения сценария. Эти специальные атрибуты обработчика событий полезны для определения сложных парадигм, в которых поток задачи изменяется в зависимости от производительности субъекта или других условий.

#### **3.2 Описание реализации**

В реализации собственных вариаций задачи «Стоп-сигнал» было использовано множество элементов (рис. 3). Далее будут рассмотрены основные элементы, отвечающие за логику программы.



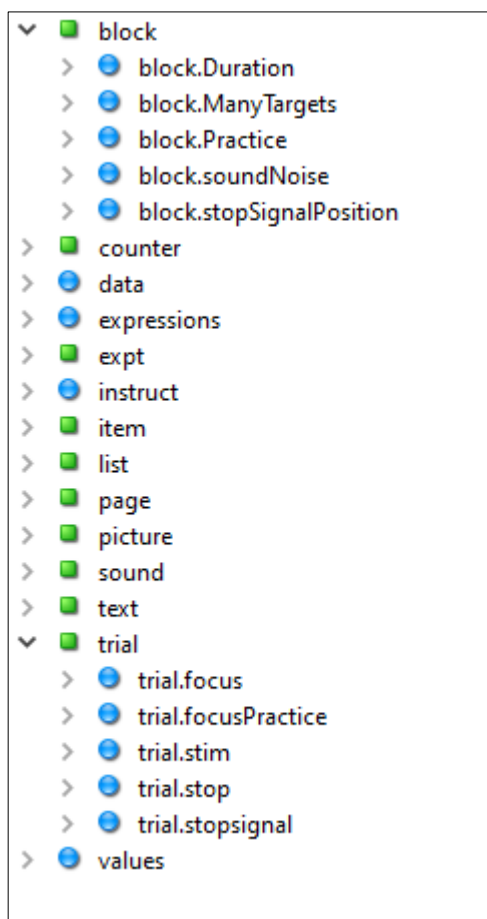


Рисунок 3 - Структура программы

Главным элементом программы является элемент `expt` (рис. 4). Этот элемент определяет последовательность запускаемых блоков (элемент `block`). В нашем случае последовательность блоков, это расположение уровней в игре. На данный момент эксперимент состоит из 6 блоков:

- 1) `block Practice` – тестовая игра;
- 2) `block ManyTargets` – это игра с дополнительными целевыми стимулами. При первом запуске блока демонстрируется уровень с двумя целевыми стимулами, при следующих запусках добавляется по одному целевому стимулу;
- 3) `block stopSignalPosition` – игра с изменением положения стоп-сигнала;
- 4) `block Duration` – игра с уменьшением времени демонстрации целевого стимула. При первом запуске блока демонстрируется уровень с временем демонстрации стимула 750мс, при каждом следующем

запуске этого блока время демонстрации целевого стимула уменьшается на 100.

5) block soundNoise – игра с фоновым шумом;

6) block Result – блок подсчёта и демонстрации результатов игры.

```
<expt >  
/blocks = [1=Practice; 2-5=ManyTargets; 6=stopPosition; 7-10= Duration; 11=Sounds; 12=Result]  
/ stop = [ (((trial.stim.totalcount) == values.practicetrialnostop)&&(values.rt_block==0))]  
</expt>
```

Рисунок 4 - Элемент expt

В элементе expt также можно указать условие завершения всего эксперимента (атрибут stop). В данной реализации, условием завершения эксперимента является случай, когда в тестовой игре испытуемый в течение первых 30-ти испытаний ни разу не отреагировал вовремя и правильно на целевой стимул. Если так произошло, то нет смысла продолжать эксперимент, так как высчитать среднее время реакции пользователя RT и время задержки стоп-сигнала SSD, не удастся.

Далее следует рассмотреть подробнее такой элемент как block. Данный элемент определяет последовательность запускаемых испытаний (trial) и инструкций. Элемент block имеет множество атрибутов, но все прописывать необязательно. В примере (рис. 5) используются атрибуты onblockbegin и onblockend. Эти атрибуты отвечают за последовательность действий, которые нужно выполнить перед началом блока или после его завершения. К примеру, подсчёт значений, изменение параметров игры или обнуление переменных.

Атрибут stop отвечает за условие завершения блока. В данном случае блок завершится, когда нужное кол-во испытаний будет продемонстрировано и пройдено, но необязательно успешно.

Атрибут skip определяет условия, при которых блок не следует запускать, а надо его пропустить. В блоках ManyTargets и Duration блок пропускается, если при предыдущем запуске испытуемым было набрано более 25% ошибок.

```

<block Duration>
/screencolor = (202, 210,187)
/onblockbegin = [values.block=values.block+1;
                 values.blockname=block.duration.name;]
/onblockend = [ list.blockname.insertitem(values.blockname,values.block);
                list.errors.insertitem(round((values.errors/values.trialnumber)*100),values.block );
                list.successtop.insertitem(round((values.successss*100)/trial.stop.count),values.block);
                list.averagert.insertitem(round(values.rt_block/trial.stim.correctcount),values.block);
                values.rt_sum=values.rt_sum+values.rt_block;
                values.rt_block=0;
                values.errors=0;
                values.succesSS=0;
                values.averagert = round(values.rt_sum/trial.stim.totalcorrectcount);
                values.stimduraction=values.stimduraction-100;
                if(values.stimduraction<450){
                    values.stimduraction=750;
                }

                if(list.errors.item(values.block)>50){
                    values.stimduraction=750;
                    values.errorflag=1;
                }
            ]
/trials = [1 = focus]
/preinstructions = (duration)
/stop = [(trial.focus.count) > values.trialnumber]
/skip = [(values.errorflag==1)|| (values.stimduraction < 450)]
</block>

```

Рисунок 5 – Блок Duration

Атрибут `preinstructions` определяет последовательность страниц инструкций, представленных в начале блока. Значением этого атрибута является описанный ранее элемент `page`. В данном элементе указывается именно текст инструкции. А параметры этого текста, например цвет или размер определяются в элементе `instruct`.

Блок состоит из испытаний (`trial`), их последовательность указывается в атрибуте блока `trials`. Элемент `trial` контролирует время и содержание предъявления стимула. В данной программе есть четыре основных элемента `trial`, отвечающие за основную логику игры:

- 1) `trial focus` – демонстрация фона в течении определенного времени (FP), определение при помощи рандомизации будет ли следующее испытание со стоп-сигналом или нет, промежуточные вычисления, переход к следующему испытанию;
- 2) `trial stim` – демонстрация целевого стимула в течении определенного времени, получение ответа и его обработка, переход обратно к `trial focus`;

3) trial stop – демонстрация целевого символа в течении времени задержки стоп-сигнала (SSD), получение ответа и его обработка, переход к trial stop signal;

4) trial stop signal – демонстрация стоп-сигнала в течение оставшегося времени получения ответа и его обработка, переход к trial focus.

Обобщим, у нас есть два вида испытаний – это испытания с появлением стоп-сигнала, реализовано в trial stop и trial stop signal, и без него, реализовано в trial stim. После каждого выполнения одного вида испытаний происходит возвращение к демонстрации фона и так по кругу, пока не сработает условие остановки блока – будет пройдено определенное количество испытаний.

Кратко устройство программы можно представить в виде схемы (рис. 6).

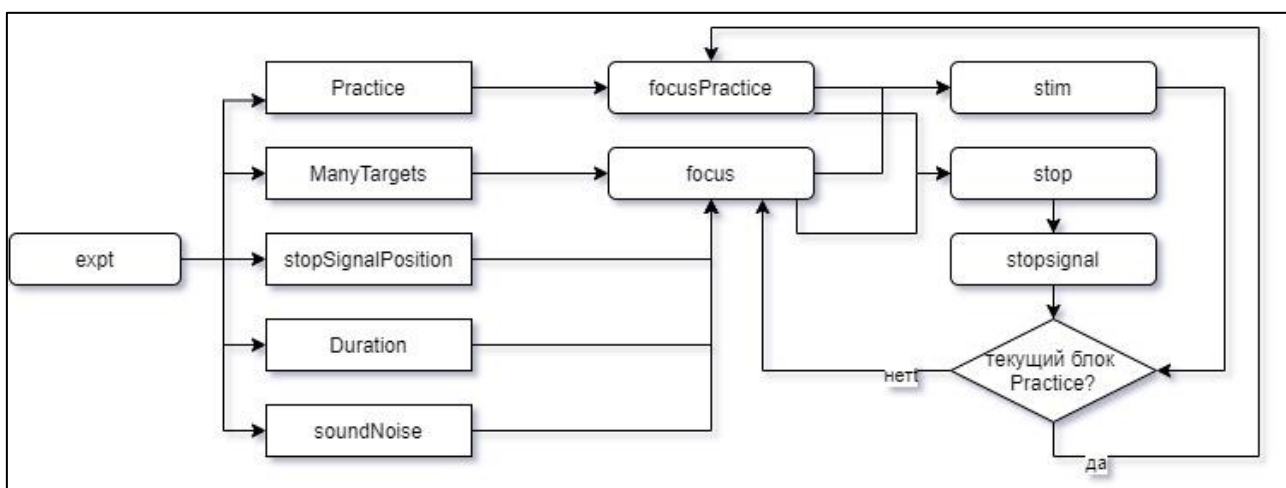


Рисунок 6 – Составляющие программы и их связи

Добавление нового блока предполагает лишь изменение значений переменных или добавление новых параметров игры, поэтому переписывать испытания для каждого блока не нужно. Если реализовать новый блок при текущей архитектуре становится сложным, тогда можно написать новое испытание и связать его с предыдущими при необходимости. Например, из-за специфичной логики в тестовой игре было создано испытание focusPractice.

После прохождения эксперимента все ответы пользователя сохраняются в файл расширения iqdat в виде таблицы (рис.7) со следующими колонками:

1) date – дата проведения эксперимента;

- 2) time – время проведения эксперимента;
- 3) group – группа испытуемого;
- 4) subject – номер испытуемого;
- 5) blockcode – название блока;
- 6) trialcode – название испытания;
- 7) picture.target.currentitem – демонстрируемый целевой стимул;
- 8) values.stiminterval – время между демонстрацией целевых стимулов;
- 9) values.stimduration – время демонстрации целевого стимула;
- 10) latency – длительность испытания;
- 11) response – ответ испытуемого на испытание;
- 12) correct – корректность ответа;
- 13) values.score – текущий счёт;
- 14) values.ssdcoef – коэффициент для вычисления SSD;
- 15) values.SSD – значение SSD;
- 16) values.averageRT – среднее время реакции;
- 17) values.errors – количество ошибок в текущем блоке.

| 0  | date   | time     | group | subject | blockcode | trialcode     | picture.target.currentitem | values.stiminterval | values.stimduration | latency | response | correct |
|----|--------|----------|-------|---------|-----------|---------------|----------------------------|---------------------|---------------------|---------|----------|---------|
| 1  | 052121 | 14:29:47 | 1     | 5       | Practice  | focusPractice | rabbit.jpg                 | 4500                | 750                 | 4500    | 0        | 0       |
| 2  | 052121 | 14:29:47 | 1     | 5       | Practice  | stim          | rabbit.jpg                 | 4500                | 750                 | 495     | 30       | 1       |
| 3  | 052121 | 14:29:47 | 1     | 5       | Practice  | focusPractice | rabbit.jpg                 | 3500                | 750                 | 3500    | 0        | 0       |
| 4  | 052121 | 14:29:47 | 1     | 5       | Practice  | stim          | tiger.jpg                  | 3500                | 750                 | 308     | 32       | 1       |
| 5  | 052121 | 14:29:47 | 1     | 5       | Practice  | focusPractice | tiger.jpg                  | 3500                | 750                 | 3500    | 0        | 0       |
| 6  | 052121 | 14:29:47 | 1     | 5       | Practice  | stim          | rabbit.jpg                 | 3500                | 750                 | 524     | 30       | 1       |
| 7  | 052121 | 14:29:47 | 1     | 5       | Practice  | focusPractice | rabbit.jpg                 | 5000                | 750                 | 5000    | 0        | 0       |
| 8  | 052121 | 14:29:47 | 1     | 5       | Practice  | stim          | tiger.jpg                  | 5000                | 750                 | 425     | 32       | 1       |
| 9  | 052121 | 14:29:47 | 1     | 5       | Practice  | focusPractice | tiger.jpg                  | 4500                | 750                 | 4500    | 0        | 0       |
| 10 | 052121 | 14:29:47 | 1     | 5       | Practice  | stim          | tiger.jpg                  | 4500                | 750                 | 410     | 32       | 1       |
| 11 | 052121 | 14:29:47 | 1     | 5       | Practice  | focusPractice | tiger.jpg                  | 3500                | 750                 | 3500    | 0        | 0       |
| 12 | 052121 | 14:29:47 | 1     | 5       | Practice  | stop          | rabbit.jpg                 | 3500                | 750                 | 221     | 0        | 1       |

Рисунок 7 – Фрагмент файла с ответами испытуемого

Вычисленные для каждого уровня характеристики испытуемого демонстрируются на экране в конце эксперимента и сохраняются в виде картинки (рис. 8) в автоматически созданную папку screenshots.

| Название блока | среднееRT | % ошибок | % успешных торможений |
|----------------|-----------|----------|-----------------------|
| Practice       | 366       | 60       | 100                   |
| ManyTargets    | 539       | 30       | 67                    |
| ManyTargets    | 561       | 20       | 100                   |
| ManyTargets    | 582       | 40       | 67                    |
| ManyTargets    | 560       | 40       | 33                    |
| stopPosition   | 0         | 90       | 50                    |
| Duration       | 520       | 30       | 100                   |
| Duration       | 527       | 40       | 33                    |
| Duration       | 457       | 40       | 67                    |
| Duration       | 412       | 40       | 50                    |
| Sounds         | 404       | 50       | 100                   |

колво набранных очков: 14  
 максимальное кол-во стимулов: 5  
 минимальное время демонстрации стимулов: 450

Рисунок 8 – Отображение результатов эксперимента

## ЗАКЛЮЧЕНИЕ

В результате работы была разработана, реализована и отлажена первая версия программы для тестирования индивидуальных способностей человека к произвольному контролю поведения для профессионального отбора персонала. Функциональное тестирование показало, что программа полностью удовлетворяет предъявляемым требованиям. Поставленные задачи полностью выполнены, а цель достигнута.

Также в рамках данной работы были составлены руководство оператора [11] (см. Приложение А) и описание программы [12] (см. Приложение Б).

В дальнейшем возможна доработка и расширение системы новыми вариациями игры «Стоп-сигнал».

Выпускная квалификационная работа выполнена мной самостоятельно и с соблюдением правил профессиональной этики. Все использованные в работе материалы и заимствованные принципиальные положения (концепции) из опубликованной научной литературы и других источников имеют ссылки на них. Я несу ответственность за приведенные данные и сделанные выводы.

Я ознакомлен с программой государственной итоговой аттестации, согласно которой обнаружение плагиата, фальсификации данных и ложного цитирования является основанием для не допуска к защите выпускной квалификационной работы и выставления оценки «неудовлетворительно».

Удод Светлана Олеговна

ФИО студента

\_\_\_\_\_  
Подпись студента

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_ г.

(заполняется от руки)

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. Logan G. D., Chambers C. D., Verbruggen F. Fictitious inhibitory differences: how skewness and slowing distort the estimation of stopping latencies. // *Psychological Science*. — SAGE Publications Inc, 2013. — Vol. 24, iss. 3. — P. 352 – 362.
2. Logan G. D., Cowan W. B. On the ability to inhibit thought and action: A theory of an act of control. // *Psychological Review*. — American Psychological Association, 1984. — Vol. 91. — P. 295–327.
3. Eagle D. M., Baunez C., Hutcheson D. M., Lehmann O., Shah A. P., Robbins T. W. Stop-Signal Reaction-Time Task Performance: Role of Prefrontal Cortex and Subthalamic Nucleus. // *Cerebral Cortex*. — Oxford University Press, 2008. — Vol. 18, iss. 1. — P. 178–188.
4. Logan G. D. On the ability to inhibit thought and action: A users' stop signal paradigm. // *Inhibitory processes in attention, memory, and language*. — CA: Academic Press, 1994. — P. 189–239.
5. Verbruggen F., Logan G. D. Models of response inhibition in the stop-signal and stop-change paradigms. // *Neuroscience & Biobehavioral Reviews*. — Elsevier, 2009. — Vol. 33. — P. 647–661.
6. Logan G. D., Schachar R. J., Tannock R. Impulsivity and inhibitory control. // *Psychological Science*. — SAGE Publications Inc, 1997. — Vol. 8. — P. 60–64.
7. Logan G. D. Attention, automaticity, and the ability to stop a speeded choice response. / In Long J., Baddeley A. D. (eds.) // *Attention and performance IX*. — Hillsdale, NJ: Erlbaum, 1981. — P. 205–222.
8. Ridderinkhof K. R., Band G. P. H., Logan G. D. A study of adaptive behavior: Effects of age and irrelevant information on the ability to inhibit one's actions. // *Acta Psychologica*. — Elsevier, 1999. — Vol. 101, iss. 2–3 — P. 315–337.



9. Verbruggen F., Liefoghe B., Vandierendonck A. The interaction between stop signal inhibition and distractor interference in the flanker and Stroop task. // Acta Psychologica. — Elsevier, 2004. — Vol. 116, iss. 1. — P. 21–37.
10. Millisecond Software. Inquisit 4 Help [Электронный ресурс]. // Inquisit Online Documentation. — Режим доступа: <https://www.millisecond.com/support/docs/v4/index.htm>, свободный.
11. ГОСТ 19.505-79. Единая система программной документации. Руководство оператора. Требования к содержанию и оформлению. // Единая система программной документации: Сборник национальных стандартов. — М.: Стандартинформ, 2010. — С. 97-99
12. ГОСТ 19.402-78. Единая система программной документации. Описание программы. // Единая система программной документации: Сборник национальных стандартов. — М.: Стандартинформ, 2010 . — С. 72-74

## **ПРИЛОЖЕНИЕ А**

### **ПРОГРАММА ДЛЯ ТЕСТИРОВАНИЯ ИНДИВИДУАЛЬНЫХ СПОСОБНОСТЕЙ К ПРОИЗВОЛЬНОМУ КОНТРОЛЮ РУКОВОДСТВО ОПЕРАТОРА**

Листов 8

Новосибирск, 2021

## СОДЕРЖАНИЕ

|  |    |
|--|----|
| Аннотация .....                                  | 28 |
| 1 Назначение программы.....                      | 29 |
| 1.1 Функциональное назначение программы .....    | 29 |
| 1.2 Эксплуатационное назначение программы .....  | 29 |
| 1.3 Состав функций.....                          | 29 |
| 2 Условия выполнения программы .....             | 30 |
| 2.1 Минимальный состав аппаратных средств .....  | 30 |
| 2.2 Минимальный состав программных средств ..... | 30 |
| 2.3 Требования к оператору .....                 | 30 |
| 3 Выполнение программы .....                     | 31 |
| 3.1 Загрузка и запуск программы .....            | 31 |
| 3.2 Выполнение программы .....                   | 31 |
| 3.3 Завершение программы .....                   | 32 |
| 4 Сообщение оператору.....                       | 33 |

## АННОТАЦИЯ

В данном документе приведено руководство оператора по применению программы для тестирования индивидуальных способностей человека к произвольному контролю поведения.

В настоящем документе, в разделе «Назначение программы» указаны сведения о назначении программы и информация, необходимая для понимания функциональности программы и ее эксплуатации. В разделе «Условия выполнения программы» перечислены условия, являющиеся необходимыми для выполнения программы. Раздел «Выполнение программы» содержит последовательность действий оператора, которые необходимы для загрузки, запуска, выполнения и завершения программы, приведено описание функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузку и управляет выполнением программы, а также ответы программы на эти команды. В разделе «Сообщения оператору» приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т.п.). Оформление программного документа «Руководство оператора» произведено по требованиям ГОСТ 19.505-79 «ЕСПД. Руководство оператора».

## **1 Назначение программы**

### **1.1 Функциональное назначение программы**

Программа на основе парадигмы «Стоп-сигнал» предназначена для тестирования индивидуальных способностей человека к произвольному контролю поведения.

### **1.2 Эксплуатационное назначение программы**

Программное обеспечение предназначено для эксплуатации в исследовательских целях. Конечными пользователями программы должны быть физические лица независимо от их рода деятельности.

### **1.3 Состав функций**

- 1) Прохождение тестовой игры «Стоп-сигнал» ;
- 2) Прохождение игры «Стоп-сигнал» на базовых настройках;
- 3) Прохождение задания «Стоп-сигнал» с тремя целевыми стимулами;
- 4) Прохождение задания «Стоп-сигнал» с четырьмя целевыми стимулами;
- 5) Прохождение задания «Стоп-сигнал» с пятью целевыми стимулами;
- 6) Прохождение задания «Стоп-сигнал» с временем демонстрации целевого стимула равным 650мс;
- 7) Прохождение задания «Стоп-сигнал» с временем демонстрации целевого стимула равным 550мс;
- 8) Прохождение задания «Стоп-сигнал» с временем демонстрации целевого стимула равным 450мс;
- 9) Прохождение задания «Стоп-сигнал» с изменением положения стоп-сигнала;
- 10) Прохождение задания «Стоп-сигнал» с фоновым шумом;
- 11) Сохранение результатов прохождения заданий.

## **2 Условия выполнения программы**

### **2.1 Минимальный состав аппаратных средств**

Для проведения эксперимента требуется следующее оборудование:

- 1) персональный компьютер для запуска программы;
- 2) клавиатура.

### **2.2 Минимальный состав программных средств**

Для работы программного обеспечения необходимы следующие программные средства:

- 1) операционная система Windows;
- 2) среда разработки Inquisit 4, установленная на компьютере для запуска программы.

### **2.3 Требования к оператору**

Конечный оператор (пользователь) программы должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы. Пользователь может не обладать специализированными знаниями в области программирования и иных научных дисциплин.

### 3 Выполнение программы

#### 3.1 Загрузка и запуск программы

Оператор нажимает на иконку программы experiment.inq после чего откроется окно (рис. А.1) для ввода номера группы и идентификатора пользователя. После ввода группы и индефикатора, для запуска эксперимента, оператор нажимает кнопку «Run».

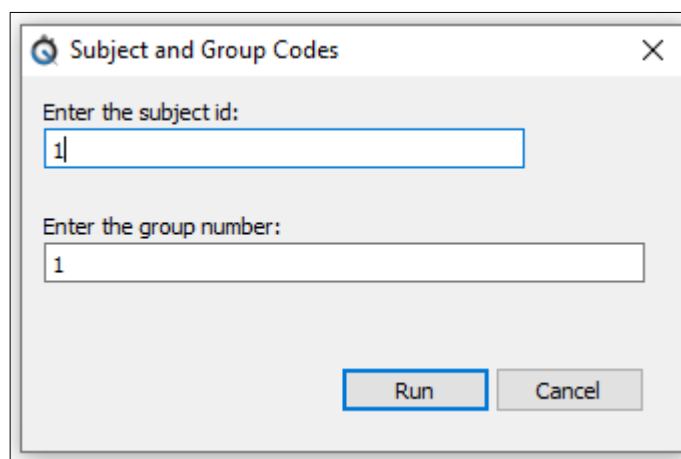


Рисунок А.1 – Окно запуска программы

#### 3.2 Выполнение программы

Программа для тестирования индивидуальных способностей к произвольному контролю представляет собой компьютерную игру, в ходе которой участники должны имитировать кормление животных. Игра содержит 11 уровней. Перед началом каждого уровня пользователю показывается текст, объясняющий правила игры. Для того чтобы перейти к игре, пользователи должны нажать клавишу Space.

На экране монитора испытуемым в случайном порядке предъявляются два визуальных стимула («тигр» или «заяц»), которых необходимо покормить (рис. А.2). Зайца морковкой (кнопка А), а тигра мясом (кнопка D). Испытуемому нужно либо нажать на нужную кнопку после появления целевого стимула, либо подавить свое уже подготовленное движение, при появлении визуального сигнала «стоп». В зависимости от уровня могут меняться количество визуальных стимулов, длительность демонстрации стимула, положение сигнала «стоп» и звук при прохождении игры.



Рисунок А.2 – Пример демонстрации визуального стимула

### 3.3 Завершение программы

Программа автоматически завершает свое выполнение после прохождения уровней игры или если тестовый уровень не был пройден. Для досрочного завершения программы нужно нажать клавиши Ctrl и Q .

После прохождения эксперимента все ответы пользователя сохраняются в файл расширения `iqdat` в виде таблицы. Это файл будет расположен в той же папке, что и сама программа. Результаты прохождения программы и характеристики пользователя сохраняются в папку `screencaptures` в виде картинки.



#### **4 Сообщение оператору**

При корректном выполнении программы, конечным пользователям достаточно той информации, что отображается на экране, а именно – правил текущего раунда и результатов игры.

## **ПРИЛОЖЕНИЕ Б**

### **СИСТЕМА ПОДДЕРЖКИ ДОКУМЕНТООБОРОТА ДЛЯ УПРАВЛЕНИЯ ЭКСПЕРТИЗЫ ПРОЕКТОВ ОПИСАНИЕ ПРОГРАММЫ**

Листов 11

Новосибирск, 2021

## СОДЕРЖАНИЕ

|   |    |
|---|----|
| Аннотация .....   | 36 |
| 1 Общие сведения.....   | 37 |
| 1.1 Обозначение и наименование программ .....                                   | 37 |
| 1.2 Программное обеспечение, необходимое для функционирования<br>программы..... | 37 |
| 1.3 Языки программирования, на которых написана программа.....                  | 37 |
| 2 Функциональное назначение .....   | 38 |
| 2.1 Назначение программы .....  | 38 |
| 2.2 Сведения о функциональных ограничениях на применение.....                   | 38 |
| 3 Описание логической структуры.....  | 39 |
| 3.1 Структура программы.....  | 39 |
| 3.2 Алгоритм программы.....   | 40 |
| 3.3 Связи программы с другими программами.....                                  | 40 |
| 4 Используемые технические средства.....  | 41 |
| 5 Вызов и загрузка.....   | 42 |
| 6 Входные данные .....  | 43 |
| 7 Выходные данные .....   | 44 |

## АННОТАЦИЯ

В данном документе приведено описание программы для тестирования индивидуальных способностей человека к произвольному контролю поведения. В настоящем документе, в разделе «Общие сведения» указаны обозначение и наименование программы, программное обеспечение, необходимое для функционирования программы, языки программирования, на которых написана программа. В разделе «Функциональное назначение» указаны классы решаемых задач. Раздел «Описание логической структуры» содержит алгоритм программы, используемые методы, структуру программы с описанием функций составных частей и связи между ними, связи программы с другими программами. Раздел «Используемые технические средства» содержит описание типов электронных вычислительных машин и устройств, которые используются при работе программы. В разделе «Вызов и загрузка» указаны способы вызова программы и входные точки в программу. В разделе «Входные данные» указаны характер, организация и предварительная подготовка входных данных, формат, описание и способ кодирования входных данных. В разделе «Выходные данные» указаны характер и организация выходных данных, формат, описание и способ кодирования выходных данных.

Оформление программного документа «Описание программы» произведено по требованиям ГОСТ 19.402-78 «ЕСПД. Описание программы» .

## **1 Общие сведения**

### **1.1 Обозначение и наименование программ**

Общие характеристики программы:

- 1) программа называется «experiment» и имеет расширение .inq;
- 2) размер исполняемого файла: 24 КБ;
- 3) язык интерфейса: русский.

### **1.2 Программное обеспечение, необходимое для функционирования программы**

Приложение для разработки и проведения психологических экспериментов Inquisit 4.

### **1.3 Языки программирования, на которых написана программа**

Исходным языком программирования является собственный синтаксис Inquisit для написания скриптов. Синтаксис выражений в целом аналогичен синтаксису языков семейства С (С, С ++, Java), в то время как другие части языка имеют поверхностное сходство с HTML или XML из-за использования элементов и атрибутов.

## **2 Функциональное назначение**

### **2.1 Назначение программы**

Программа на основе парадигмы «Стоп-сигнал» предназначена для тестирования индивидуальных способностей человека к произвольному контролю поведения.

При помощи данной программной системы работодатель сможет оценивать сотрудников на то, удовлетворяют ли их способности определенным требованиям, и на основе полученных характеристик сможет подобрать подчиненному подходящую работу.

### **2.2 Сведения о функциональных ограничениях на применение**

Все ограничения упираются в ограничения используемых аппаратных и программных средств.

### 3 Описание логической структуры

#### 3.1 Структура программы

Главным элементом программы является элемент expt. Этот элемент определяет последовательность запускаемых блоков. В нашем случае последовательность блоков, это расположение уровней в игре. На данный момент эксперимент состоит из 6 блоков:

- 1) block Practice – тестовая игра;
- 2) block ManyTargets – это игра с дополнительными целевыми стимулами. При первом запуске блока демонстрируется уровень с двумя целевыми стимулами, при следующих запусках добавляется по одному целевому стимулу;
- 3) block stopSignalPosition – игра с изменением положения стоп-сигнала;
- 4) block Duration – игра с уменьшением времени демонстрации целевого стимула. При первом запуске блока демонстрируется уровень с временем демонстрации стимула 750мс, при каждом следующем запуске этого блока время демонстрации целевого стимула уменьшается на 100.
- 5) block soundNoise – игра с фоновым шумом;
- 6) block Result – блок подсчёта и демонстрации результатов игры.

Устройство программы можно представить в виде схемы (рис. Б.1).

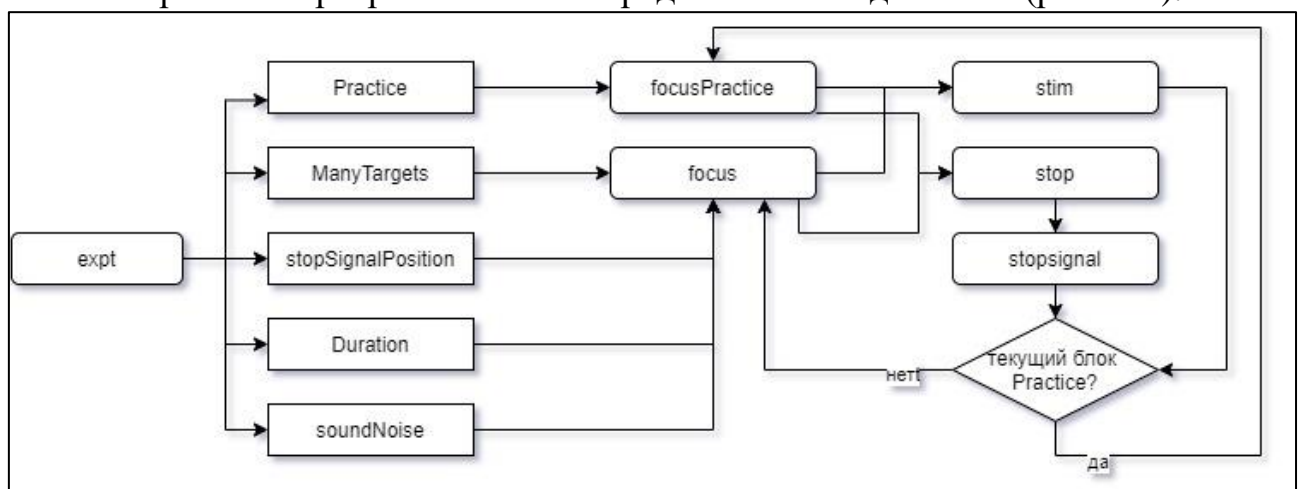


Рисунок Б.1 – Составляющие программы и их связи

### 3.2 Алгоритм программы

Алгоритм работы программы представлен на рисунке Б.2 в виде блок-схемы.

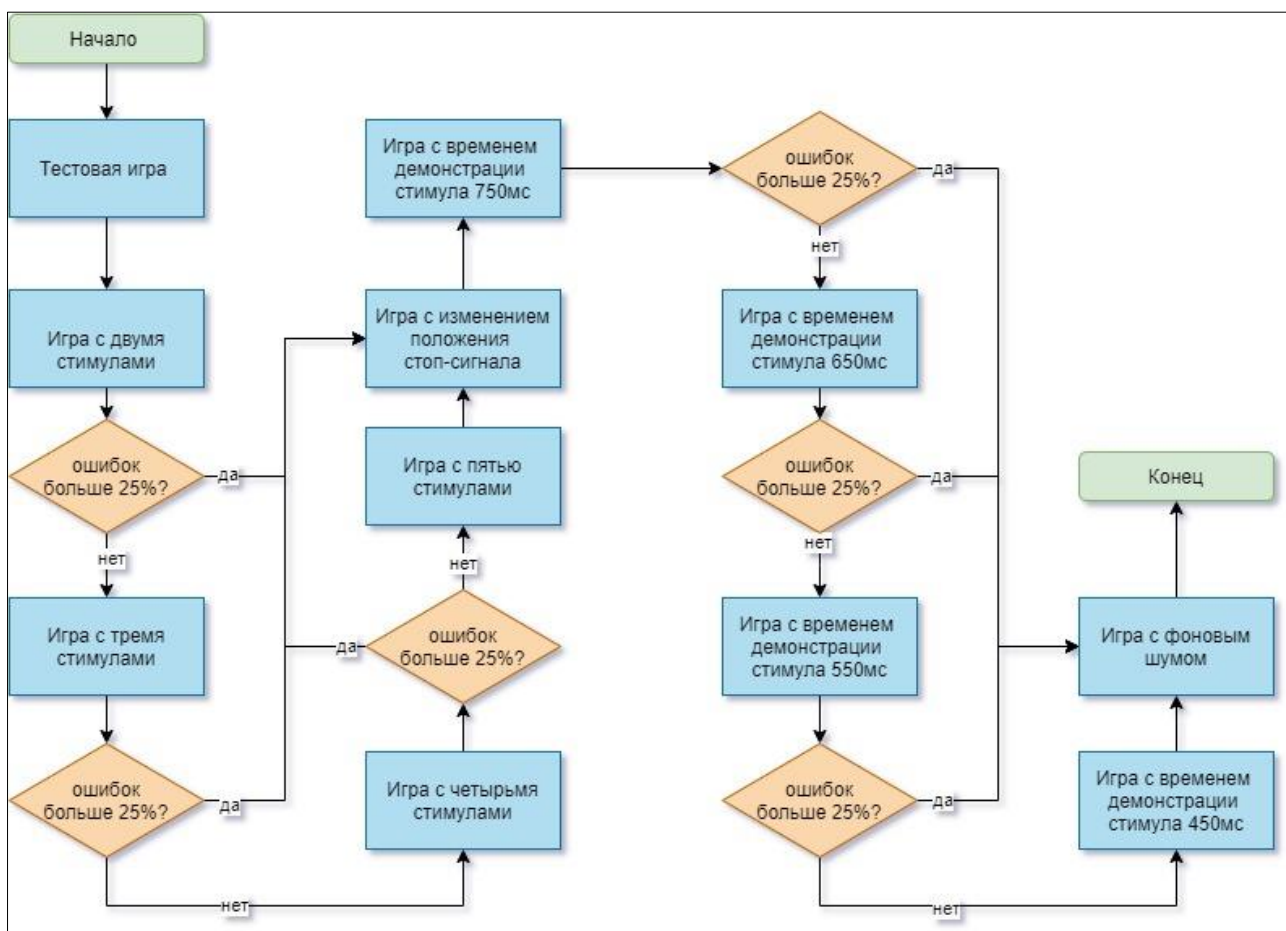


Рисунок Б.2 – Алгоритм работы программы

### 3.3 Связи программы с другими программами

Программа запускается на основе программы Inquisit, поэтому для ее функционирования необходима предустановленная программа Inquisit (версии 4).



#### **4 Используемые технические средства**

Для проведения эксперимента требуется следующее оборудование:

- 1) персональный компьютер для запуска программы;
- 2) клавиатура.

Для работы программного обеспечения необходимы следующие программные средства:

- 1) операционная система Windows;
- 2) среда разработки Inquisit 4, установленная на компьютере для запуска программы.

## **5 Вызов и загрузка**

Оператор нажимает на иконку программы `experiment.inq` после чего откроется окно для ввода номера группы и идентификатора пользователя. После ввода группы и идентификатора, для запуска эксперимента, оператор нажимает на кнопку «Run».

## **6 Входные данные**

Входные данные указываются в исходном коде программы в элементах `item` и `value`.

## **7 Выходные данные**

После прохождения эксперимента все ответы пользователя сохранятся в файл расширения `iqdat` в виде таблицы. Это файл будет расположен в той же папке, что и сама программа. Результаты прохождения программы и характеристики пользователя сохранятся в папку `screencaptures` в виде картинки.