



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Московский технологический университет»

МИРЭА

Институт

Информационных технологий

наименование института (полностью)

Кафедра

Вычислительной техники

наименование кафедры (полностью)

КУРСОВЫЕ РАБОТ ПО ДИСЦИПЛИНЕ

Объектно-ориентированное программирование

(наименование дисциплины)

Направление подготовки

(код и наименование)

Профиль

(код и наименование)

Форма обучения

очное

(очная, очно-заочная, заочная)

Программа подготовки

(академический, прикладной бакалавриат)

Квалификация выпускника

Бакалавр

Москва 2018

Содержание

1.	введение.....	3
1.1.	Игровой цикл.....	5
1.2.	Отчет по курсовой работе	7
2.	угадай число.....	8
3.	Угадай число компьютер.....	8
4.	Угадай слово («Словомеска»).....	8
5.	игра «поле чудес».....	9
6.	Игра «Крестики-нолики».....	9
7.	Игра «15».....	10

1. ВВЕДЕНИЕ

Лабораторные работы построены в определенной последовательности и являются расширением функционала предыдущей версии.

Исходная постановка и соответствующий код приведен в методике выполнения лабораторных работ.

0	<p><u>Проверка готовности объектов к работе.</u></p> <p>Описание базового класса объектов cl_base.</p> <p>Описание класса приложения.</p> <p>Алгоритм обхода иерархического дерева.</p> <p>main</p> <p>Конструирование иерархии объектов.</p> <p>exes_app</p> <p>Проверка готовности объектов к работе.</p>
1	<p><u>Вывод на консоль иерархии объектов.</u></p> <p>Добавить в класс cl_base:</p> <ul style="list-style-type: none">- метод обхода иерархии объектов и вывода наименований объектов в виде дерева (show_object_next).- метод show_object_tree вызова метода show_object_next для текущего объекта. <p>main</p> <p>Конструирование иерархии объектов.</p> <p>exes_app</p> <p>Вывод на консоль иерархии объектов.</p>

2	<p><u>Доступ к объектам по координатам на дереве иерархии.</u></p> <p>Добавить в класс <code>cl_base</code>:</p> <ul style="list-style-type: none"> - метод получения наименования объекта по индексу из координаты объекта; - метод получения ссылки на корневой объект; - метод получения ссылки от корневого объекта согласно координате. <p><code>main</code> Конструирование иерархии объектов.</p> <p><code>exes_app</code> Доступ к трем объектам по координатам на дереве иерархии и вывод их наименований на консоль.</p>
3	<p><u>Ввод, обработка и вывод данных.</u></p> <p>Расширения функционала в трех пользовательских классах.</p> <ol style="list-style-type: none"> 1. Добавить метод ввода массива целых чисел. 2. Добавить метод упорядочения массива. 3. Добавить метод вывода массива на консоль. <p><code>main</code> Конструирование иерархии объектов.</p> <p><code>exes_app</code> Построение алгоритма решения задачи посредством последовательного вызова выше приведенных методов объектов.</p>

4	<p><u>Сигналы и обработчики.</u></p> <p>Добавить в класс <code>cl_base</code> методы и фрагменты кода согласно лекционному курсу:</p> <ul style="list-style-type: none"> - макрорасширения с параметрами; - объявление ассоциативного списка для хранения ссылок на методы сигналов и обработчиков согласно построенным пользователем интерфейсов между объектами; - метод построения интерфейса сигнал-обработчик; - метод удаления интерфейса сигнал-обработчик; - метод выдачи сигнала. <p><code>main</code> Конструирование иерархии объектов. Построение интерфейсов между объектами.</p> <p><code>exes_app</code> Реализация алгоритма решения задачи посредством цикла ввода и выдачи определенных сигналов.</p>
<u>Курсовая работа</u>	
К	<p>Курсовая работа реализует алгоритм определенной диалоговой игры. Программа строиться согласно правилам построения объектно-ориентированного программирования.</p> <p>Для всех задействованных объектов использовать класс <code>cl_base</code> в качестве базового.</p> <p><code>main</code> Конструирование иерархии объектов. Построение интерфейсов между объектами.</p> <p><code>exes_app</code> Запуск основного игрового цикла.</p>

1.1. Игровой цикл

Игровой цикл – это обобщенное представление потока событий, происходящих в игре.

Реализация цикла от игры к игре может существенно различаться, фундаментальная структура цикла является практически одинаковой во всех играх любых жанров. Игровой процесс обычно можно разбить на одинаковые повторяющиеся компоненты, образующие игровой цикл.

Этапы игрового цикла.

Установка параметров. На данном этапе игра принимает исходные параметры или загружает игровые ресурсы (картинки, звуки, музыку и т.д.). Пользователю предлагается ознакомиться с правилами игры и ее целью.

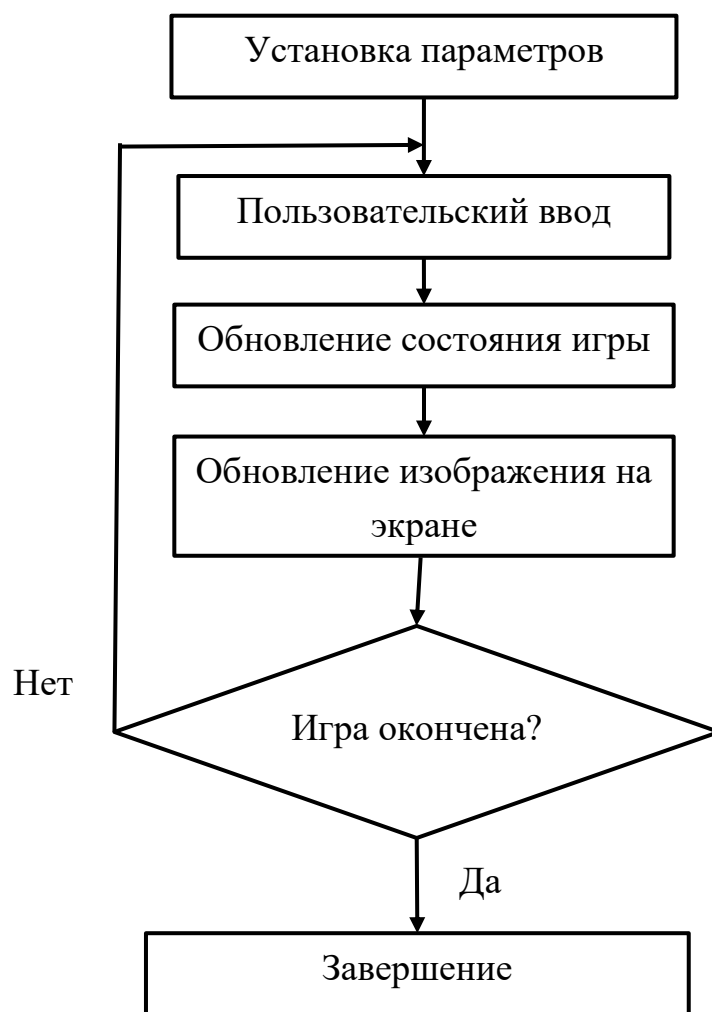
Получение пользовательского ввода. Программа принимает пользовательский ввод через то или иное устройство ввода: клавиатура, мышь, джойстик и т.д.

Обновление внутреннего состояния игры. Отрабатывают игровые правила и логика, при этом учитывается полученный ранее пользовательский ввод. Отрабатывают правила взаимодействия объектов. Проводятся вычисления «искусственным интеллектом» виртуального противника.

Обновление изображения на экране. Изображение на экране однозначно отображает текущее состояние игры. Важно при реализации обратить особое внимание на полноту и корректность отображаемой информации.

Проверка того, не окончена ли игра. Если игра не окончена, то цикл повторяется с этапа пользовательского ввода данных. Если игра окончена, то происходит переход к завершающему этапу.

Завершение. На этом этапе игра завершена. Пользователю предоставляется определенная итоговая информация.



При проектировании программы курсовой работы, каждый этап игрового цикла реализовать в отдельном классе.

1.2. Отчет по курсовой работе

Содержательно форма отчета по курсовой работе совпадает с формой отчета по лабораторной работе. Меняется только наименование: «Отчет по курсовой работе».

2. УГАДАЙ ЧИСЛО

Компьютер выбирает случайное число в диапазоне от 1 до N. Игрок пытается угадать это число с наименьшим количеством попыток. После ввода очередного значения, компьютер сообщает игроку, насколько близко это число расположено к загаданному: слишком много, слишком мало или почти совпал. Если пользователь угадает число, игра заканчивается.

Параметры игры:

N – правая граница интервала для выбора случайного числа.

После завершения игры выдается количество попыток.

Предусмотреть команду принудительного завершения игры. Тогда пользователю сообщается, сколько попыток он сделал и какое случайное число было отобрано.

3. КОМПЬЮТЕР, УГАДАЙ ЧИСЛО

Написать версию программы «Угадай число», где компьютер и игрок меняются ролями: игрок выбирает число, а компьютер должен его угадать. Игрок задает близость предложенного компьютером варианта к загаданному числу. Программа должна отслеживать корректность ответов пользователя, сообщая о неверном определении близости предложенного компьютером варианта с загаданным числом.

Параметры игры:

N – правая граница интервала для выбора случайного числа.

После завершения игры выдается количество попыток.

Предусмотреть команду принудительного завершения игры. Тогда на консоль выводиться, сколько попыток он сделал компьютер и какое случайное число было отобрано пользователем.

4. УГАДАЙ СЛОВО («СЛОВОМЕСКА»)

По команде пользователя «Слово», компьютер загадывает слово, перемешивает его буквы случайным образом предоставляет пользователю.

Пользователь должен угадать слово. Допускается произвольное количество попыток. Если пользователь затрудняется, то можно попросить у компьютера подсказку.

Игра завершается по команде «Конец».

Пользователю начисляются очки согласно количеству букв угаданного слова. За каждую дополнительную попытку (другую версию слова) очки снимаются. Если просит подсказку, очки снимаются.

При завершении игры вывести: какое максимальное количество баллов можно было набрать, набранное количество очков и количество угаданных слов.

5. ИГРА «ПОЛЕ ЧУДЕС»

Компьютер загадывает слово, которое пользователь пытается отгадать по буквам. Игрок имеет право на N ошибок. Если слово не угадано, а N ошибок допущены, то игра проиграна.

Загаданное компьютером слово на консоли отобразить вопросительными знаками «?», согласно количеству букв в слове. В ходе игры вопросительный знак меняется на угаданную букву.

Количество ошибок N является параметром игры и вводится игроком. Значение по умолчанию 8.

6. ИГРА «КРЕСТИКИ-НОЛИКИ»

Реализовать игру «Крестики-нолики».

Первоначально пользователь определяет количество поединков (серия игр). Поединки играют подряд. В начале серии пользователь определяет, кто начинает первый поединок. он или компьютер. Тот, кто начинает играет крестиками. В последующих поединках серии право первого хода чередуется.

Первоначально на консоли вывести пустые клетки.

```
  |   |  
---+---+---  
  |   |  
---+---+---  
  |   |
```

Свой ход пользователь задает номером ячейки согласно следующей нумерации:

0	1	2
3	4	5
6	7	8

Необходимо выставить одинаковые знаки по горизонтали, по вертикали или по диагонали. Выигравший получает 2 очка, проигравший 0 очков. При ничейном результате даются по одному очку. В конце серии определяется победитель.

7. ИГРА «15»

Реализовать игру «15».

Первоначально компьютер формирует случайное распределение чисел от 1 до 15 в таблице 4x4. Передвигая соседние с пустой клеткой таблицы числа необходимо добиться их последовательного расположения в таблице. В верхнем левом углу таблицы должен стоять 1 и далее по порядку.

Случайное распределение должен быть допустимым для решения задачи.

Правила перемещения чисел:

A – влево;

W – вверх;

D – вправо;

Z – вниз.

Если требуемая последовательность собрана, игра завершается. В конце игры выдается количество сделанных ходов.

Предусмотреть команду принудительного завершения игры. При этом тоже выдается количество сделанных ходов.

Финальное, выигрышное расположение чисел

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	