Министерство образования и науки Российской Федерации

ФГАОУ ВО «Уральский Федеральный Университет

имени первого Президента России Б. Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РтФ

Оценка

\_\_\_\_\_\_\_\_\_\_\_\_

**СОЗДАНИЕ WEB-ПРИЛОЖЕНИЯ “BLACKJACK” С ПОМОЩЬЮ РАЗЛИЧНЫХ МЕТОДОВ МАТЕМАТИКИ**   
пояснительная записка к курсовому проекту по модулю

«Современные сетевые технологии»

Дата Подпись

Преподаватель \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ Корелин И. А.

Студент \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ Борутя Д. А.

Студент \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ Ветошкин А. А.

Группа РИ-450005

Екатеринбург

2019

**ОГЛАВЛЕНИЕ**

ВВЕДЕНИЕ………………………………………………………………………..3

1. BLACKJACK……………………………………………………………….4
2. ХОД ИГРЫ…………………………………………………………………5
3. ГЕНЕРАТОРЫ СЛУЧАЙНЫХ ЧИСЕЛ………………………………….6
   1. Равномерное распределение………………………………………..6
   2. Треугольный закон распределения………………………………...8
4. СОЗДАНИЕ ПРИЛОЖЕНИЯ…………………………………………….10

ЗАКЛЮЧЕНИЕ………………………………………………………..………...12

БИБЛИОГРАФИЧЕСКИЙ СПИСОК………………………………………..…13

ПРИЛОЖЕНИЕ А. Blackjack.html……………………………………………...14

ПРИЛОЖЕНИЕ Б. Blackjack.css………………………………………………..15

ПРИЛОЖЕНИЕ В. Blackjack.js…………………………………………………18

**ВВЕДЕНИЕ**

Целью выполнения курсового проекта является приобретение практических навыков в современных сетевых технологиях. Создания Web-приложения “Blackjack” с помощью различных методов математики.

В соответствии с поставленной целью можно выделить следующие задачи:

1. Составление плана разработки приложения;
2. Поиск подходящей литературы;
3. Разработка Web-приложения;
4. Тестирование приложение;
5. **BLACKJACK**

Blackjack — одна из самых популярных карточных игр в казино по всему миру. Большая популярность игры обуславливается простыми правилами, скоростью игры и наиболее простой стратегией в подсчёте карт. Тем не менее популярность игра завоевала не сразу. Игорным домам США приходилось стимулировать интерес к игре различными видами бонусов и выработкой нескольких разновидностей правил для блек-джека. Считается, что предшественником этой игры была карточная игра «vingt-et-un» («двадцать один»), которая появилась во французских игорных заведениях приблизительно в 19 веке. В России, например, блек-джек по сей день часто называют двадцать одно или очко.

**Цель игры в блэк-джек.**

В блэк-джек играют против дилера. Поэтому основная цель заключается в том, чтобы собрать комбинацию карт, превосходящую руку дилера, но без перебора. Лучшая рука имеет 21 очко. Любая комбинация, сумма очков которой больше 21 автоматически выбывает из игры.

Обратите внимание, не нужно всегда набирать 21 очко для того, чтобы выиграть у дилера. Для этого достаточно иметь руку, сумма очков которой, превышает общую сумму очков у дилера.

**Значения карт в блэк-джек.**

Все карты с лицами (король, дама и валет) приносят 10 очков. Туз может стоить как 1, так и 11 очков, в зависимости от того, какое значение выгоднее в данный момент. Стоимость остальных карт равна их числовому значению. Например, 3 пик принесет вам 3 очка, а 6 треф - 6 очков. Масти карт никакого отношения к их стоимости не имеет.

1. **ХОД ИГРЫ**
2. Дилер перемешивает все колоды карт.
3. Игроки делают ставки.
4. Дилер всем игрокам раздает по 2 карты, в том числе и себе и при этом открывает свою одну карту. Есть вариант также, что всем раздают по две карты, а дилер себе оставляет только одну открытую. После чего себе набирает карты.
5. Игроки оценивают свои карты и открытую карту дилера.
6. После оценки ситуации игроки могут взять любое количество карт или остаться на той сумме очков что есть. Главное сумма очков не должно превышать «21» очко.
7. Набор карт игроками проходит строго по очереди.
8. После того как все набрали карты или не взяли в зависимости от ситуации — дилер открывает вторую карту.
9. После открытия второй карты дилер обязан по правилам брать карты если у него 16 и меньше очков и остановиться когда у него будет 17 и больше очков.
10. Игроки свои карты не имеют право трогать — открывает их только дилер.
11. **ГЕНЕРАТОРЫ СЛУЧАЙНЫХ ЧИСЕЛ**
12. **Равномерное распределение.**

В основе метода Монте-Карло лежит генерация случайных чисел, которые должны быть равномерно распределены в интервале (0; 1).

Если генератор выдает числа, смещенные в какую-то часть интервала (одни числа выпадают чаще других), то результат решения задачи, решаемой статистическим методом, может оказаться неверным. Поэтому проблема использования хорошего генератора действительно случайных и действительно равномерно распределенных чисел стоит очень остро.

Математическое ожидание m и дисперсия D такой последовательности (рис. 1), состоящей из n случайных чисел r[i], должны быть следующими (если это действительно равномерно распределенные случайные числа в интервале от 0 до 1):



Рис. 1 Мат. Ожидание и Дисперсия

Если пользователю потребуется, чтобы случайное число x находилось в интервале (a; b), отличном от (0; 1), нужно воспользоваться формулой x = a + (b – a) · r, где r — случайное число из интервала (0; 1). Законность данного преобразования демонстрируется (рис. 2).



Рис. 2

Теперь x — случайное число, равномерно распределенное в диапазоне от a до b.

За эталон генератора случайных чисел (ГСЧ) принят такой генератор, который порождает последовательность случайных чисел с равномерным законом распределения в интервале (0; 1). За одно обращение данный генератор возвращает одно случайное число. Если наблюдать такой ГСЧ достаточно длительное время, то окажется, что, например, в каждый из десяти интервалов (0; 0.1),(0.1; 0.2), (0.2; 0.3), …, (0.9; 1) попадет практически одинаковое количество случайных чисел — то есть они будут распределены равномерно по всему интервалу (0; 1). Если изобразить на графике k = 10 интервалов и частоты N[i] попаданий в них, то получится экспериментальная кривая плотности распределения случайных чисел (рис. 3).



Рис. 3 Частотная диаграмма.

1. **Треугольный закон распределения (Закон Симпсона).**

Он характерен для случайных погрешностей цифровых приборов, в которых измеряемая величина преобразуется в пропорциональный интервал времени Тсч, называемый временем счета, а измерение этого интервала выполняется с помощью счетных импульсов стабильного генератора, имеющих период следования Т0. В связи со случайным положением счетных импульсов относительно интервала Тсч, а также случайным соотношением между периодом Т0 и временем счета Тсч треугольный закон представляет собой композицию (объединение) двух равномерных законов с одинаковыми по величине максимальными погрешностями.

Функция распределения одномерной плотности вероятности случайных погрешностей для треугольного закона задается следующими соотношениями (рис. 4):



Рис. 4.

Где mx, a - параметры распределения.

График треугольного закона распределения приведен на рисунке 5.

Математическое ожидание величины x: определяется по той же формуле, что и равномерное:

.



Рис. 5 Треугольное распределение случайной величины.

Среднее квадратическое отклонение определяется по формуле:

**.**

1. **СОЗДАНИЕ ПРИЛОЖЕНИЯ**

В ходе создания Web-приложения “Blackjack” были использованы технология разработки такие как:

1. HTML;
2. CSS;
3. JavaScript.

Был создан дизайн приложения с помощью HTML и CSS (рис. 6, 7).



Рис. 6 Игровое поле

С помощью языка JavaScript была написана логика приложения, отвечающая за выполнения различных функций, методов и решений.



Рис. 7 Процесс игры

Для алгоритма взятия карты из колоды были использованы два алгоритма генерации случайных величин:

1. Равномерное распределения – для выбора масти карты;
2. Треугольный закон распределения – для выбора номинала карты.

**ЗАКЛЮЧЕНИЕ**

В результате выполнения данного курсового проекта было создано Web-приложение “Blackjack” с помощью различных методов математики и логики.

В ходе выполнения работы был составлен план выполнения работы, найдена подходящая литература, использованы различные методы решения, использованы разные генераторы случайных чисел для взятия карты игроку.

Была создана логика Web-приложения, которая отвечала за весь функционал игры. Был создан интерфейс окна игры.

**БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. <https://ru.wikipedia.org/wiki/Блэкджек> // [Электронный ресурс]
2. <http://add-hobby.ru/blackjack.html> // [Электронный ресурс]
3. <http://stratum.ac.ru/education/textbooks/modelir/lection22.html> // [Электронный ресурс]
4. <http://lib.kstu.kz:8300/tb/books/Metrologiya/teory/246.html> // [Электронный ресурс]

**ПРИЛОЖЕНИЕ А. Blackjack.html**

<!DOCTYPE html>

<html lang = "en">

<head>

<meta charset="utf-8">

<link rel="stylesheet" href="blackjack.css">

</head>

<body>

<div class="game" >

<h3 >Blackjack</h3>

<div class="game-body">

<div class="game-options" >

<input type="button" id="btnStart" class="btn" value="Начать" onclick="startblackjack()">

<input type="button" id="STOP" class="btn" value="Взять еще" onclick="hitMe()">

<input type="button" id="STOP1" class="btn" value="Хватит" onclick="stay()">

</div>

<div class="status" id="status" ></div>

<div id="deckcount" class="deck" >52</div>

<div id="next\_card" class="next" ></div>

<div id="players" class="players" > </div>

</div>

</div>

</body>

<script src="blackjack.js" ></script>

</html>

**ПРИЛОЖЕНИЕ Б. Blackjack.css**

.game

{

width: 720px;

border: solid ;

margin: 0 auto;

background: green;

border: solid 10px;

text-align:center;

min-height:630px;

}

.btn{

border:none;

padding: 10px 30px;

border-radius:10px;

cursor:pointer;

}

.btn:hover{

background:dodgerblue;

}

.game h3

{

padding: 10px;

background:black;

color:white;

}

.game .game-body

{

padding: 20px;

}

.game .game-options

{

text-align:center;

}

.game .card{

width: 50px;

height: 80px;

padding: 10px;

background-color: white;

display: inline-block;

border-radius: 10px;

font-size: 10pt;

text-align: center;

margin: 3px;

border:solid 3px;

}

.game .status

{

display: inline-block;

background: gold;

border: solid 5px white;

border-radius: 10px;

padding: 5px 20px;

margin: 0 auto;

margin-top: 10px;

display:none;

}

.players{

width: 500px;

margin: 0 auto;

margin-top: 20px;

text-align: center;

}

.player{

width: 70%;

border: solid 5px white;

border-radius: 10px;

padding: 10px;

display: inline-block;

margin:1%;

vertical-align: top;

background:white;

}

.player.active{

border: solid 10px black;

}

.points{

text-align: center;

font-size: 20pt;

font-weight: bold;

margin: 10px;

}

.game .deck{

display: inline-block;

float:left;

background: white;

color:black;

padding: 15px;

border-radius:15px;

border: double 5px;

display: none;

}

.game .next{

display: inline-block;

background: white;

color: black;

border-radius: 50%;

padding: 15px;

margin-top: 70px;

margin-left: -50px;

float: left;

display: none;

}

.btn

{

background:black;

color:white !important;

border-radius:0px;

}

**ПРИЛОЖЕНИЕ В. Blackjack.js**

/\*eslint-env browser\*/

var suits = ["Spades", "Hearts", "Diamonds", "Clubs"];

var values = ["3","5","7","9","K","Q","10","J","A","8","6","4","2"]

var deck = new Array();

var players = new Array();

var currentPlayer = 0;

function createDeck()

{

deck = new Array();

for (var i = 0 ; i < values.length; i++)

{

for(var x = 0; x < suits.length; x++)

{

var weight = parseInt(values[i]);

if (values[i] == "J" || values[i] == "Q" || values[i] == "K")

weight = 10;

if (values[i] == "A")

weight = 11;

var card =

{

Value: values[i],

Suit: suits[x],

Weight: weight

};

deck.push(card);

}

}

}

function uniform\_random()

{

var result = 1 + Math.floor(Math.random() \* (suits.length));

return result;

}

function triangular\_random()

{

var result = Math.floor((1 + Math.floor(Math.random() \* (values.length)) + 1 + Math.floor(Math.random() \* (values.length)))/2);

return result;

}

function hitMe()

{

var card\_suits = uniform\_random();

var card\_values = triangular\_random();

var weight = parseInt(values[card\_values]);

if (values[card\_values] == "J" || values[card\_values] == "Q" || values[card\_values] == "K")

weight = 10;

if (values[card\_values] == "A")

weight = 11;

var card =

{

Value: values[card\_values],

Suit: suits[card\_suits],

Weight: weight

};

var test\_deck = new Array();

test\_deck.push(card);

var count = 0

for(var i = 0; i< deck.length; i++)

{

if(deck[i].Suit == test\_deck[0].Suit && deck[i].Value == test\_deck[0].Value)

{

var temp = deck[i];

deck[i] = deck[deck.length-1];

deck[deck.length-1] = temp;

var card\_del = deck.pop();

players[currentPlayer].Hand.push(card\_del);

renderCard(card\_del, currentPlayer);

updatePoints();

updateDeck();

nextcard();

check();

count++;

break;

}

}

if (count==0)

{

hitMe();

}

}

function nextcard()

{

var card\_values = triangular\_random();

var weight = values[card\_values];

document.getElementById('next\_card').innerHTML = weight;

document.getElementById('next\_card').style.display = "inline-block";

}

function first\_cards()

{

for(var i = 0;i<2;i++)

{

hitMe();

}

currentPlayer++;

hitMe();

currentPlayer--;

}

function bot()

{

}

function createPlayers(num)

{

players = new Array();

for(var i = 1; i <= num; i++)

{

var hand = new Array();

var player =

{

Name: 'Player ' + i,

ID: i,

Points: 0,

Hand: hand

};

players.push(player);

}

}

function startblackjack()

{

document.getElementById("STOP").disabled = false;

document.getElementById("STOP1").disabled = false;

document.getElementById('btnStart').value = 'Заново';

document.getElementById("status").style.display="none";

currentPlayer = 0;

createDeck();

createPlayers(2);

createPlayersUI();

document.getElementById('deckcount').style.display = "inline-block";

first\_cards();

document.getElementById('player\_' + currentPlayer).classList.add('active');

nextcard();

check();

}

function createPlayersUI()

{

document.getElementById('players').innerHTML = '';

for(var i = 0; i < players.length; i++)

{

var div\_player = document.createElement('div');

var div\_playerid = document.createElement('div');

var div\_hand = document.createElement('div');

var div\_points = document.createElement('div');

div\_points.className = 'points';

div\_points.id = 'points\_' + i;

div\_player.id = 'player\_' + i;

div\_player.className = 'player';

div\_hand.id = 'hand\_' + i;

div\_playerid.innerHTML = 'Player ' + players[i].ID;

div\_player.appendChild(div\_playerid);

div\_player.appendChild(div\_hand);

div\_player.appendChild(div\_points);

document.getElementById('players').appendChild(div\_player);

}

}

function renderCard(card, player)

{

var hand = document.getElementById('hand\_' + player);

hand.appendChild(getCardUI(card));

}

function getCardUI(card)

{

var el = document.createElement('div');

var icon = '';

if (card.Suit == 'Hearts')

icon='&hearts;';

else if (card.Suit == 'Spades')

icon = '&spades;';

else if (card.Suit == 'Diamonds')

icon = '&diams;';

else

icon = '&clubs;';

el.className = 'card';

el.innerHTML = card.Value + '<br/>' + icon;

return el;

}

function getPoints(player)

{

var points = 0;

var ace = 0;

for(var i = 0; i < players[player].Hand.length; i++)

{

points += players[player].Hand[i].Weight;

if(players[player].Hand[i].Weight == 11)

ace++;

}

if(ace>0 && points>21)

points-=10;

players[player].Points = points;

return points;

}

function updatePoints()

{

for (var i = 0 ; i < players.length; i++)

{

getPoints(i);

document.getElementById('points\_' + i).innerHTML = players[i].Points;

}

}

function stay()

{

if (currentPlayer != players.length-1)

{

document.getElementById('player\_' + currentPlayer).classList.remove('active');

currentPlayer += 1;

document.getElementById('player\_' + currentPlayer).classList.add('active');

hitMe();

}

else {

end();

}

}

function end()

{

var winner = -1;

var score = 0;

for(var i = 0; i < players.length; i++)

{

if (players[i].Points > score && players[i].Points < 22)

{

winner = i;

}

score = players[i].Points;

}

document.getElementById('status').innerHTML = 'Выиграл: Player ' + players[winner].ID;

document.getElementById("status").style.display = "inline-block";

document.getElementById("STOP").disabled = true;

document.getElementById("STOP1").disabled = true;

}

function check()

{

if (players[currentPlayer].Points > 21)

{

document.getElementById('status').innerHTML = 'Player: ' + players[currentPlayer].ID + ' проиграл';

document.getElementById('status').style.display = "inline-block";

document.getElementById("STOP").disabled = true;

document.getElementById("STOP1").disabled = true;

end();

}

else if(players[currentPlayer].Points == 21)

{

end();

}

}

function updateDeck()

{

document.getElementById('deckcount').innerHTML = deck.length;

}