

**Заключение.** Благодаря социальным сетям компании могут легко донести информацию о своих продуктах и брендах до широкой аудитории, создавать целевые рекламные кампании и взаимодействовать с клиентами.

Социальные сети существенно влияют на принятие потребителями решений о покупках, формировании их предпочтений и лояльности к брендам. Поэтому компаниям важно учитывать этот фактор при разработке маркетинговых стратегий и взаимодействии с аудиторией.

#### Список цитируемых источников

1. Статистика социальных сетей в мире (сентябрь 2024) : [сайт]. — URL: <https://inclient.ru> (дата обращения 28.09.2024).
2. Самые популярные социальные сети в мире по состоянию на апрель 2024 года, по количеству активных пользователей в месяц потребителя : [сайт]. — URL: <https://www.statista.com> (дата обращения 28.09.2024).
3. Сколько людей пользуются социальными сетями (статистика 2024 года) : [сайт]. — URL: <https://www.demandsage.com> (дата обращения 28.09.2024).
4. Статистика доходов и использования Facebook за 2024 год: конкурентные тенденции, которые вам нужно знать : [сайт]. — URL: <https://digixxe.com> (дата обращения 28.09.2024).
5. Как Instagram зарабатывает деньги : [сайт]. — URL: <https://www.investopedia.com> (дата обращения 28.09.2024).

УДК 004.9

**Д. В. Гордич**

*Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь*

*Научный руководитель  
А. И. Калько*

### МОДЕЛИРОВАНИЕ ЛОГИЧЕСКОЙ ИГРОВОЙ ПРОГРАММЫ «BRAIN BUSTERS» СРЕДСТВАМИ VISUAL STUDIO C++

**Введение.** Компьютер служит помощником в решении задач и развлечением. Игровая индустрия активно развивается уже полвека, притягивая внимание к разработке игровых вселенных, их проектированию и реализации. Актуальность данной курсовой работы заключается в возможности попробовать свои силы в этой сфере.

Цель работы — разработка логической игры «Brain Busters» или «Угадай слово». Задача — освоить возможности Visual Studio C++ для создания игр.

Visual Studio C++ — это наиболее продвинутая визуальная среда для быстрой разработки на C++ под Windows, содержащая около 200 компонентов, что облегчает создание программ. В работе использовалась именно эта среда [1].

Visual Studio интегрирует Палитру компонентов, разделённую на функциональные группы, и предлагает библиотеку из более чем 100 повторно используемых визуальных компонентов, которые можно перетаскивать на форму. Кроме стандартных элементов управления Windows, библиотека включает новые компоненты для диалогов и работы с базами данных [2].

Уникальное сочетание удобства разработки пользовательских интерфейсов и компонентной архитектуры упрощает программирование в Visual Studio [3].

**Основная часть.** Компьютер выбирает случайное слово из списка, добавляет в него несколько случайных букв меняет буквы местами и выводит на экран. Задача игрока — угадать слово за наименьшее время. Игра должна содержать меню, используя которое можно:

- 1) устанавливать уровень сложности в зависимости от длины слова (до 5 букв лёгкий, до 8 средний, больше 8 тяжёлый);
- 2) ознакомиться с правилами игры получить основную информацию о программе;
- 3) таблица рекордов.

Для начала определим главную часть программы «Brain Busters», которая заключается в угадывании слова, которое изменяется с помощью компьютера. Логично предположить, что сначала игроку будет тяжело втянуться в процесс, поэтому форм, в которых угадываются слова будет 3, конечно же, каждая из них соответствует своему уровню сложности. Так же в каждой новой форме добавляется новый усложняющий игру фактор, а именно: добавление букв или символов в слово, таймер.

Программа позволяет угадывать заранее записанные в файл слова с последующей зашифровкой их, путём преобразований изнутри и добавлении ложных символов (в зависимости от уровня сложности). Основной функционал достаточно прост, он состоит из текстовых полей и кнопок, отслеживающих результата.

Преобразование слов и их последующее разгадывание достаточно интересный процесс. Сначала мы должны вызвать случайное число, чтобы после из файла достать то слово, что соответствует этому числу, после мы благодаря библиотеке IO указываем имя файла:

“WordSpisok.txt” — текстовый файл, где хранятся слова, которые в будущем нужно будет угадывать игроку.

Как его нужно открыть и какое слово и из какой строки его стоит взять. Когда мы взяли нужное слово настало время преобразовать его в некоторый шифр, в слове буквы поменяются местами используя цикл for, в нём мы каждую букву слова заносим в массив, а после при помощи функции rand() меняем их местами.

“CharArray[i]” — массив, каждый элемент которого является буквой этого слова.

Дальше используем команду “Swap()” (сортировка пузырьком) для конечного преобразования слова. Также можно добавить после выхода из цикла при помощи «+» пару других символов, чтобы запутать игрока [4].

Все элементы программы для свободного перемещения по программе расположены функциональном окне. Вид функционального окна показан на рисунке 1. По нажатию кнопки «Правила» открывается функциональное окно, в котором можно поясняются правила игры. Вид функционального окна показан на рисунке 2.



Рисунок 1 — Вид главного функционального окна

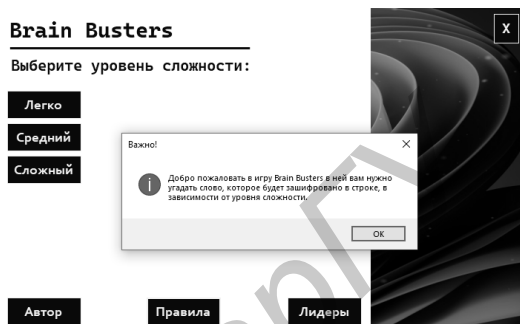


Рисунок 2 — Вид главного функционального окна и результат нажатия на кнопку «Правила»

Далее выбрав уровень сложности (в нашем случае) «Сложный» открывается новая форма со своим интерфейсом. По нажатию на кнопку «Важно!» откроется функциональное окно, в котором подробно распишут сложности уровня. Вид функционального окна показан на рисунке 3. После нажатия кнопки «Начать» на этом уровне сложности попросят ввести своё имя для записи вас в список лидеров (рисунок 4). Для того чтобы быть на первом месте на этом уровне сложности вы должны отгадать загаданное слова за минимально потраченное время. Когда вы отгадаете слово вам откроется функциональное окно, которое похвалит вас за выполненную работу (рисунок 5). Сверху идёт таймер, который передаст ваши результаты в таблицу.



Рисунок 3 — Вид функционального окна «Сложный» и нажатие на кнопку «Важно!»

Игрок	Время
1 ББББ	0:13
2 Дмитрий	0:17
3 дмриа	0:17
4 Олег	0:25
5 Антон	0:26
6 Егор	0:31
7 Меля	0:35
8 дерба	0:39
9 деее	0:42
10 Игорь	0:45
11 Иуиц	0:46
12 Лева	0:51
13 Игорь	0:51
14 Никита	0:57
15 цуцц	1:22
16 кцук	1:55

Рисунок 4 — Таблица лидеров «Brain Busters» уровня Сложный



Рисунок 5 — Функциональное окно победы в уровне и последующих поздравлений

**Заключение.** Разработка программы «Brain Busters» требует умственных усилий и знаний C++. Важными этапами являются описание спецификации и алгоритма, несмотря на упрощения, предоставляемые C++. Тестирование подтвердило работоспособность программы с небольшими улучшениями. Выявленные недостатки предполагают дальнейшую работу над совершенствованием, что предоставляет начинающему программисту возможность повысить знания и навыки. Улучшение программы потребует изменений в исходном коде. Программа может использоваться для проведения свободного времени.

#### Список цитируемых источников

1. Visual studio 2022 и его графические компоненты : [сайт]. — 2024. — URL: <https://learn.microsoft.com/ru-ru/visualstudio/profiling/memory-usage?view=vs-2022> (дата обращения: 26.03.2024).
2. Функции Windows GDI+ : [сайт]. — 2024. — URL: <https://learn.microsoft.com/ru-ru/windows/win32/gdiplus/-gdiplus-gdi-start> (дата обращения: 26.03.2024).
3. Свойство Graphics в Visual Studio 2022 : [сайт]. — 2024. — URL: <https://learn.microsoft.com/ru-ru/visualstudio/designers/image-editor?view=vs-2022> (дата обращения: 26.03.2024).
4. Мазур, С. А. Разработка приложения “SQUARES” в среде разработки Unity / С. А. Мазур, С. Н. Шапутько, А. И. Калько // Техника и технологии: инновации и качество : материалы V Междунар. науч.-практ. конф. (Барановичи, 20 дек. 2018 г.) / М-во образования Респ. Беларусь, Баранович. гос. ун-т ; [редкол.: В. В. Климук (гл. ред.) и др.]. — Барановичи, 2019. — С. 15–16.

УДК 519.683

Д. В. Гордич, Е. И. Дулько

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

Научный руководитель

Ю. П. Нерода

## КАК МАТЕМАТИКА ПОМОГАЕТ СОЗДАВАТЬ СЛУЧАЙНЫЕ ЧИСЛА В КОМПЬЮТЕРЕ

**Введение.** В математике, программировании и технике часто требуется создавать видимость случайности, например, для чисел или изображений. Хотя они кажутся случайными, на самом деле они создаются с помощью специальных алгоритмов. Компьютеры постоянно генерируют такие числа, даже если не обмениваются данными, чтобы эффективно распределять программы в памяти. Однако, будучи предсказуемыми машинами, компьютеры не могут создавать настоящие случайные числа. Вместо этого они используют псевдослучайные числа. Называют так числа потому, что генерируют их определенные алгоритмы. Рассмотрим простые методы создания таких псевдослучайных чисел.

**Основная часть.** Определение того, что именно является случайностью, может быть довольно сложной задачей. Существуют тесты (например, колмогоровская сложность), которые могут дать точное значение того, насколько случайна та или иная последовательность.

Попробуем создать последовательность чисел, которые будут казаться несвязанными между собой. Часто требуется не просто одно число, а несколько случайных чисел, генерируемых непрерывно. Следовательно, учитывая начальное значение, нам нужно создать другие случайные числа. Это начальное значение называется семенем. Seed (семя) — основа псевдослучайных алгоритмов.

Один из подходов может заключаться в том, чтобы применить какую-то безумную математическую формулу к семени, а затем исказить её настолько, что число на выходе будет казаться непредсказуемым, а после взять его как семя для следующей итерации. Такой подход называется линейным конгруэнтным методом. Он очень часто используется, потому что он прост в реализации и вычисления выполняются быстро. Суть метода заключается в вычислении последовательности случайных чисел  $X$  полагая:

$$X_{n+1} = (a \cdot X_n + c) \bmod m,$$

где  $X_n$  — это семя;

$\bmod$  — остаток от деления;

$a$  и  $c$  — константы [1].

В разных языках программирования реализация линейного конгруэнтного метода отличается, то есть меняются значения констант. Например, функция случайных чисел в библиотеке языка программирования C использует  $m = 2^{32}$ ,  $a = 1664525$  и  $c = 1013904223$ .

Чтобы показать на примере принцип работы этого метода, была создана программа для визуализации подкидывания игрального кубика и монетки. Эта логическая программа была написана на языке программирования C++ в среде VISUAL STUDIO [2].