Учреждение образования

«Белорусский государственный технологический университет»

**Кафедра информационных систем и технологий**

**«Отчёт по лабораторной работе №12»**

“Исследование стеганографического метода на основе преобразования наименее значащих битов”

**Выполнил:** студент 4 курса

1 группы специальности ИСИт

Палазник Арсений Викторович

**Проверил:** преподаватель

Сазонова Дарья Владимировна

Минск 2024

**Разработать собственное приложение, в котором должен быть реализован метод НЗБ. При этом:**

* **выбор файла-контейнера – по согласованию с преподавателем;**
* **реализовать два варианта осаждаемого/извлекаемого сообщения: − собственные фамилия, имя и отчество; − текстовая часть отчета по одной из выполненных лабораторных работ;**
* **реализовать два метода (на собственный выбор) размещения битового потока осаждаемого сообщения по содержимому контейнера;**
* **сформировать цветовые матрицы (по аналогии с рис. 12.7), отображающие каждый задействованный для осаждения уровень младших значащих битов контейнера;**
* **выполнить визуальный анализ (с привлечением коллег в качестве экспертов) стеганоконтейнеров с различным внутренним содержанием; сделать выводы на основе выполненного анализа.**

Реализация приложения:

|  |
| --- |
| <script>  const uploadImage = document.getElementById('uploadImage');  const canvas = document.getElementById('canvas');  const ctx = canvas.getContext('2d');  const messageBox = document.getElementById('message');  const output = document.getElementById('output');  let image = new Image();  // Загрузка изображения  uploadImage.addEventListener('change', (e) => {  const file = e.target.files[0];  const reader = new FileReader();  reader.onload = (event) => {  image.src = event.target.result;  };  reader.readAsDataURL(file);  });  // Отрисовка изображения на canvas  image.onload = () => {  canvas.width = image.width;  canvas.height = image.height;  ctx.drawImage(image, 0, 0);  };  // Скрытие сообщения в младших битах  document.getElementById('hideMessage').addEventListener('click', () => {  const message = messageBox.value;  if (!message) {  alert('Введите сообщение для скрытия');  return;  }  const imgData = ctx.getImageData(0, 0, canvas.width, canvas.height);  const data = imgData.data;  // Кодирование сообщения  const binaryMessage = message  .split('')  .map(char => char.charCodeAt(0).toString(2).padStart(8, '0'))  .join('') + '00000000'; // Стоп-бит  let messageIndex = 0;  for (let i = 0; i < data.length && messageIndex < binaryMessage.length; i += 4) {  const binaryPixel = data[i].toString(2).padStart(8, '0');  const newPixel = binaryPixel.slice(0, -1) + binaryMessage[messageIndex];  data[i] = parseInt(newPixel, 2);  messageIndex++;  }  ctx.putImageData(imgData, 0, 0);  alert('Сообщение скрыто в изображении');  });  // Извлечение сообщения  document.getElementById('revealMessage').addEventListener('click', () => {  const imgData = ctx.getImageData(0, 0, canvas.width, canvas.height);  const data = imgData.data;  let binaryMessage = '';  for (let i = 0; i < data.length; i += 4) {  const binaryPixel = data[i].toString(2).padStart(8, '0');  binaryMessage += binaryPixel.slice(-1);  }  const chars = binaryMessage.match(/.{8}/g);  let message = '';  for (const char of chars) {  const code = parseInt(char, 2);  if (code === 0) break; // Стоп-бит  message += String.fromCharCode(code);  }  output.textContent = `Извлечённое сообщение: ${message}`;  });  </script> |

Результат выполнения:





