**ЛАБОРАТОРНА РОБОТА №7**

*Анімація JavaScript і DOM*

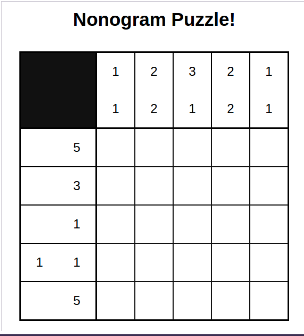
***Мета роботи:*** вивчити засоби по роботі з об’єктною моделлю документа; навчитися додавати, змінювати та видаляти анімації на вебсторінках; засвоїти принципи роботи з CSS-стилями.

**Порядок виконання роботи**

1. Ознайомитися із теоретичними відомостями.
2. Виконати завдання до лабораторної роботи.
3. Скласти та оформити звіт.

**Теоретичні відомості**

[Японські кросворди](https://en.wikipedia.org/wiki/Nonogram#See_also) (також відомі як "Пі-крос", "Ханьцзе", "Числова сітка" і т.д.) – це акуратні маленькі логічні головоломки, які швидко вивчаються, а також легко програмуються. Хоча вам не потрібно знати, як вони працюють, щоб виконати цю лабораторну роботу, ви можете дізнатися, як їх розв'язувати на рисунку нижче.



**Задання до роботи**

Для виконання цієї лабораторної роботи Вам надаються файли HTML (nonograms.html) та CSS (nonograms.css) із початковим кодом. Вам потрібно буде лише трохи змінити кожен з цих файлів для декількох завдань роботи. В іншому випадку ви просто напишете код JavaScript у новому файлі з назвою nonograms.js.

**Примітка**: У нас також є інші нонограмні пазли більших розмірів (10x10, 15x15, 20x20), якщо ви хочете використовувати їх замість маленьких. Ці HTML-файли використовують той самий файл CSS.

Приклад готової головоломки можна знайти [тут](https://courses.cs.washington.edu/courses/cse154/18sp/18sp-data/lab/lab04/code/solution/nonograms.html).

**Завдання I: Створення спливаючого вікна, коли користувач клацає на комірці**

Спочатку ми потренуємося додавати обробники подій до колекції існуючих DOM-елементів. Напишіть JavaScript-файл nonograms.js, який налаштовує обробники подій під час завантаження сторінки.

Зокрема, він повинен прикріпити функцію до кожної комірки в сітці (яка має клас .box), щоб коли користувач натискає на цю комірку, з'являлося спливаюче повідомлення "Ви натиснули на комірку!".

**Завдання II: Зробити так, щоб окрема комірка ставала чорною при натисканні**

Друге завдання полягає в тому, щоб додати на сторінку функціонал, який би заповнював окрему комірку, коли користувач натискає на неї. Існує клас .filled, який обробляє стилізацію для вас в CSS. Вам слід додати цей клас до комірки, на яку натискають, щоб вона заповнилася.

**Завдання III: Реалізація перемикання заливки**

Тепер, коли ви додали функціонал для заповнення окремих комірок пазлу, змініть обробник події кліку для кожної комірки так, щоб коли користувач клацає на комірці, то вона заповнювалася, якщо вона біла, інакше вона очищалася (ставала білою), якщо вона заповнена.

Не існує спеціального класу для "незаповнених" комірок. Ви можете очистити заповнену комірку, просто видаливши клас .filled.

**Завдання IV: Додавання ефектів наведення**

Тепер, коли ви зробили так, що користувач може заповнювати і очищати кожну комірку, додайте код, який підсвічує межі кожної комірки, на яку наведено курсор.

Зверніть увагу, що для цього насправді не потрібен JavaScript! CSS3 дозволяє нам додавати псевдо-селектори до елементів на сторінці. :hover тут особливо корисний. Трохи змініть CSS так, щоб будь-яка плитка (з класом box), на яку наведено курсор, мала box-shadow 0px 0px 2px 2px 2px #111111 (без відстані по горизонталі/вертикалі, але з розмиттям у 2px і розкидом у 2px).

**Завдання V: Додавання кнопки "Очистити" на сторінку**

Іноді ви просто робите багато помилок і хочете почати все спочатку. Вручну клацати кожну заповнену комірку (або оновлювати сторінку), щоб очистити пазл, може бути неприємно. Додайте до nonograms.html кнопку з написом "Очистити", яка очистить всі заповнені комірки, видаливши всі класи .filled. За бажанням, ви можете додати підтвердження в цей обробник події, щоб користувач мав можливість двічі подумати, перш ніж очистити всю свою важку роботу!

**Завдання VI: Перетягніть і заповніть**

Тепер, коли ви можете заповнювати кілька комірок, чи не хочете ви оптимізувати швидкість вирішення? Додайте функціональність до сторінки, щоб коли користувач клацав комірку, утримував її та перетягував через сітку комірок, комірка, яку він натиснув, перемикала клас .filled, а будь-яка комірка на шляху змінювалася, щоб мати той самий стан, що й обрана комірка (після її заливки/очищення).

Іншими словами, якщо комірка, яку було клацнуто, була заповнена, її слід очистити, а потім будь-яку комірку, передану під час перетягування користувачем миші (поки він не зніме палець з кнопки миші), також слід очистити, якщо вона вже не є.

Якщо користувач клацає за межами сітки комірок, а потім перетягує свою мишу в сітку комірок, перш ніж відпустити кнопку миші, комірки не змінюватимуться.

**Завдання VII: Впровадження X-Filling**

Багато розв’язувачів нонограм люблять позначати комірки, які, як вони знають, не будуть заповнені «х». Це полегшує швидкий перегляд комірок, які залишилися, які можна заповнити.

Додайте клас під назвою crossed-out до small-grid.css, який дає «X» як фонове зображення на заповненій комірці щоразу, коли її клацають, використовуючи цю URL-адресу фонового зображення: https://courses.cs.washington.edu/ courses/cse154/18sp/18sp-data/lab/lab04/slides/hidden/img/x.png. Зауважте, що розміри цього файлу .png становлять 256 x 256 пікселів, тому ви повинні використовувати значення 100% для background-size, коли встановлюєте фонове зображення.

**Проблеми виклику**

Гарна робота! Тепер у вас є повнофункціональна головоломка з нонограмою! Існує ряд інших функцій, які ви можете додати, щоб потренуватися працювати з DOM, анімаціями, стилями тощо. Ось кілька ідей, якщо ви маєте бажання потренуватися:

1. Дозвольте користувачеві перетягувати «х» по сторінці. Тобто відтворюйте ефект перетягування, який ви зробили для заповнення/очищення комірок, але перемикайте лише «заповнені x» комірки, коли користувач клацає правою кнопкою миші замість лівої.
2. Дозвольте користувачеві вибрати, яким кольором він хоче заповнити комірки, додавши на сторінку спадне меню.
3. Перевірте головоломку! Виявляється, для цього не так багато роботи. Додайте на сторінку кнопку «Надіслати», щоб, коли користувач натискає її, на сторінці з’являлося повідомлення про успішне вирішення головоломки.
4. Напишіть функції JavaScript, які генерують сітку голограми з нуля! Ви маєте всі необхідні класи у файлі CSS. Ви можете змінити CSS або додати власні класи, щоб створити сітку того самого розміру (15 x 15 фрагментів) або іншого розміру (5 x 5, 10 x 10, 10 x 15, 25 x 25 тощо)