

**Дата: 05.09.2023**

**Група: 33**

**Предмет: Технології**

## **УРОК 2**

**ТЕМА: «Колірні моделі та палітри. Формати графічних файлів. Методи стискання та кодування графічних даних»**

### **МЕТА:**

- Вивчити поняття «колірних моделей»
- Опанувати основні відомості про формати графічних файлів
- Розвинути навички роботи з методами стискання та кодування графічних файлів
- Виховати інформаційно-освічену особу, цікавість до обраної професії, дисципліну та уважність

**ОБЛАДНАННЯ:** ПК, підручник, плакати, електронна презентація.

**ТИП УРОКУ:** Вивчення нового матеріалу.

### **ХІД УРОКУ:**

Сьогодні на уроці ми опануємо основи організації кольорознавства та його корекцію. Основна причина, з якій доводиться виконувати корекцію кольору, наступна: людське око має здатність адаптуватися до сили і спектральним характеристикам освітлення таким чином, що зберігається сприйняття кольору предметів в більшості випадків незалежно від спектрального складу освітлення, камера же фіксує світлове випромінювання без адаптації. Так при перегляді в інших умовах, фотографії іноді сильно відрізняються від того, що ми бачили, коли фотографували. Для усунення цієї проблеми в фотографії використовуються алгоритми вибору і настройки білого кольору. Ці алгоритми вже можна назвати кольорокорекцією.

### **1. Колірні моделі.**

Згадаємо, що кольори пікселів растрових зображень кодуються числами. Так само кодуються кольори об'єктів у векторних зображеннях і будь-які інші кольори в комп'ютерній графіці. Механізм, за яким певному числу або набору чисел зіставляється колір, визначається **колірною моделлю**.

**Колірною моделлю називають систему кодування кольорів, яка використовується для зберігання, відображення на екрані та друку зображення.**

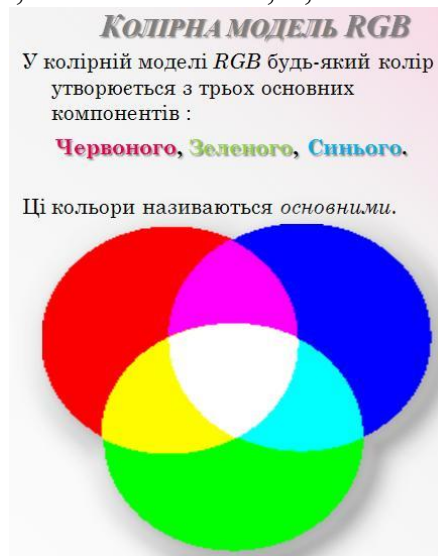
Є десятки колірних моделей, більш чи менш поширених. Ми розглянемо три найуживаніші — RGB, CMYK і HSB.

### **Колірна модель RGB.**

Для того щоб ми змогли що-небудь побачити, потрібне світло. Його випромінюють сонце, електричні лампи, вогонь, газова горілка, телевізор, монітор. Світло може мати

певний колір, а світло, що здається нам безколірним, насправді містить кольорові складові (згадайте радугу). Кольори джерел світла можна описати за допомогою колірної моделі RGB, у якій кольори розглядаються як результат змішування (додавання) трьох базових кольорів: **червоного (Red)**, **зеленого (Green)**, **синього (Blue)**.

Частки базових кольорів визначаються трьома числами з діапазону від 0 до 255 та записуються в порядку їх згадування в моделі. Так, чистий червоний колір у моделі RGB подається як 255, 0, 0 (червона складова максимальна, зелена і синя — відсутні), чистий зелений — як 0, 255, 0, а синій — як 0, 0, 255



Чорний колір подається як 0, 0, 0 (жоден з кольорів не випромінюється, частка всіх складових дорівнює нулю). Білий колір відповідає максимуму випромінювання — рівень кожної складової максимальний; у числовому вигляді білий колір записується так: 255, 255, 255.

Позаяк частка кожного базового кольору описується 256 числами, всього в моделі RGB можна описати  $256^3$  кольорів (близько 16,8 мільйона).

### Колірна модель СМҮК.

Ви задумувалися над тим, чому об'єкт, на який падає світло, має певний колір? Це пояснюється тим, що матеріал поверхні об'єкта однікомпоненти світла поглинає, а інші відбиває. Якщо, наприклад, об'єкт відбиває лише червоні компоненти світла, вони сприймаються оком, і об'єкт здається нам червоним.

Для опису цього явища застосовують колірну модель СМҮК (Cyan, Magenta, Yellow, Black), базові кольори якої утворено шляхом вилучення з білого світла базових кольорів моделі RGB: **блакитний** (білий мінус червоний), **пурпуровий** (білий мінус зелений), **жовтий** (білий мінус синій).

Основна сфера застосування колірної моделі СМҮК — повно колірний друк, оскільки у разі її використання за допомогою чотирьох фарб можна отримати на папері повноколірне зображення. Це стає можливим завдяки властивості людського ока «усереднювати» колір маленьких і близько розташованих об'єктів. Тому якщо, наприклад, дуже близько або з накладанням нанести на папір пурпурні та жовті

крапки, око сприйматиме їх як область червоного кольору. Друкуючи у такий спосіб крапки чотирьох базових кольорів, можна отримати більшість існуючих кольорів.



Колір у моделі CMYK задається чотирма числами від 0 до 100, які визначають інтенсивність (yellow) і 5 % чорного (black). Цьому кольору відповідає четвірка чисел (30, 45, 80, 5). Білий колір — це повна відсутність кольору (значення всіх колірних складових дорівнюють 0), а чорний утворюється, коли значення усіх компонентів максимальні. Змішування трьох компонентів CMY в рівних кількостях дає відтінки сірого. Хоча моделі RGB і CMYK взаємопов'язані, не всі кольори моделі CMYK можуть бути подані в моделі RGB і навпаки: найяскравіші кольори моделі RGB неможливо передати за допомогою CMYK, а для найтемніших кольорів моделі CMYK немає аналогів у RGB.

## Колірна модель HSB

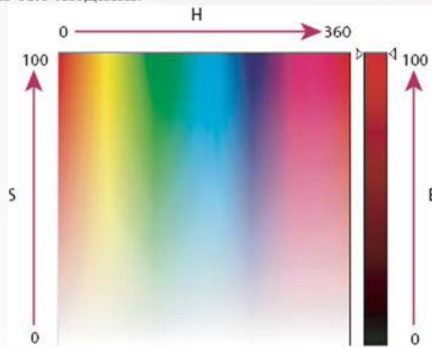
Розглянуті колірні моделі базуються на змішуванні основних кольорів. Але коли ми бачимо певний колір, нам складно сказати, скільки в ньому відсотків червоного, зеленого і синього, проте легко визначити колірний тон, насиченість та яскравість.

## КОЛІРНА МОДЕЛЬ HSB

У моделі HSB також використовуються 3 базових компонента: *відтінок*, *контраст* або *насиченість* і *яскравість* кольору.

**Відтінок** кольору вказує номер кольору в спекторній панелі. **Насиченість** кольору характеризує його інтенсивність - чим вона більша, тим "Чистіший" колір.

**Яскравість** кольору залежить від домішки чорної фарби до даного кольору - чим більше, тим яскравість кольору менша. Таким чином, можна отримувати всі кольори, які здатне сприймати око людини.



Третя модель, яку називають **HSB** (Hue, Saturation, Brightness) або **HSV** (Hue, Saturation, Value), враховує цей аспект та визначає колір за допомогою трійки чисел, кожне з яких відповідає одній з базових його характеристик.

- **Відтінок (Hue)** задає розташування кольору у спектрі видимого світла, тобто визначає, яким саме є колір: жовтим, червоним, синім тощо. Значення 0 колірного тону відповідає червоному кольору, який змінюється на жовтий, потім — на зелений, блакитний, синій, пурпуровий і знову червоний.
- **Насиченість (Saturation)** визначає, наскільки колір є інтенсивним, віддаленим від сірого. Нульова насиченість відповідає відтінку сірого кольору.
- **Яскравість (Brightness)** визначає, наскільки колір світлий чи темний. Нульова яскравість — чорний колір, максимальна — білий.

### Палітра кольорів.

Палітра вказує поточний Колір 1 (колір переднього плану) та поточний Колір 2 (колір тла). Їх використання залежить від виконуваних у програмі дій.

Використовуючи палітру, можна виконати певні дії (табл. 1).

Таблиця 1. Можливості використання палітри

Можливості використання палітри	Дії
Змінювати вибраний колір переднього плану	На вкладці Початок у групі Кольори натиснути кнопку Колір 1, а потім обрати колір
Змінювати вибраний колір фону	На вкладці Початок у групі Кольори натиснути Колір 2, а потім обрати колір
Малювати вибраним кольором переднього плану	Переміщувати покажчик обраною фігурою, утримуючи ліву клавішу миші
Малювати вибраним кольором фону	Натиснути праву клавішу миші і утримувати її під час переміщення покажчика
Встановлювати поточний колір переднього плану	На вкладці Початок у групі Інструменти натиснути кнопку Вибір кольору і вибрати на зображенні колір, який буде задано як колір переднього плану

Встановлювати колір фону	На вкладці Початок у групі Інструменти натиснути кнопку Вибір кольору і правою клавішею миші клацнути колір на зображенні, який буде задано як колір фону
Заповнювати кольором усе зображення або замкнену фігуру	На вкладці Початок у групі Інструменти клацнути кнопку Заливка; у групі Кольори натиснути Колір 1, вибрати колір, а потім клацнути усередині області, щоб залити її вибраним кольором
Змінювати палітру для вибору нового кольору	На вкладці Початок у групі Зображення клацнути кнопку Зміна палітри; у діалоговому вікні Зміна палітри клацнути колір у палітрі, а потім натиснути кнопку ОК. Колір з'явиться в одній із комірок палітри, після чого з ним можна буде працювати в Paint

## 2. Кольорокорекція.

«Кольорокорекція» — внесення змін до кольору оригіналу.

Корекція кольору не є ретушню фотографій і не зможе виправити ефект «червоних очей».

Основна причина, з якої доводиться виконувати корекцію кольору, наступна: людське око має здатність адаптуватися до сили і спектральним характеристикам освітлення таким чином, що зберігається сприйняття кольору предметів в більшості випадків незалежно від спектрального складу освітлення, камера же фіксує світлове випромінювання без адаптації. Так при перегляді в інших умовах, фотографії іноді сильно відрізняються від того, що ми бачили, коли фотографували. Для усунення цієї проблеми в фотографії використовуються алгоритми вибору і настройки білого кольору. Ці алгоритми вже можна назвати кольорокорекцією.

Інші причини застосування кольорокорекції: недостатній або надлишковий контраст зображення, вуаль, вицвітання зображення. Також необхідно відзначити кольорокорекцію з метою внесення гармонії між фотозображеннями і дизайном публікації або вмістом матеріалів, подібну процесу фотодруку в фотографічному мистецтві.

Як і будь-який інший метод, корекцію кольору можна застосувати як інструмент реалізації творчого задуму чи побажань замовника.

Методи перетворення кольору можуть бути самими різними, проте найбільш часто використовуваними методами кольорокорекції є наступні:

Вказівка аргументів для функцій перетворення вхідних значень у вихідні. В програмах ці інструменти називаються «рівні», «гамма» і т. д. Іноді набори значень аргументів вибираються з переліку наперед заданих варіантів. До цього виду перетворень можна віднести так само:

Установка балансу білого (облік освітлення) при перетворенні електронного сигналу матриці в файл зображення або скануванні плівки;

Перетворення, безпосередньо що задають зміни контрастності, яскравості, гама, тони, Безпосереднє завдання графіків перетворення значень по каналам. Цей інструмент зазвичай називається «Криві» (Curves). Він дозволяє виконати будь-які перетворення всередині кожного каналу шляхом ручного формування графіка, аналогічного тим, які обчислюються функціональними алгоритмами по заданих аргументам.

Тепер розповімо про те, що саме ми робимо з фотографіями в плані корегування кольору.

Наш сервіс використовує при обробці фотографій професійний відкалібрований монітор, який показує всі кольори природно, без спотворень. Безпосередньо для кольорокорекції використовуємо напівавтоматичний режим колірної корекції -

програмно-ручний спосіб. Тобто в спеціальному професійному графічному редакторі включаємо режим автоматичної корекції і, після того як комп'ютер закінчить свою роботу над фотографіями, доводимо колір кожної фотографії вручну до ідеалу.

### 3. Формати графічних файлів.

**Графічні формати файлів** і даних призначені для зберігання зображень, таких як фотографії та малюнки.

Графічні формати поділяться на векторні і растрові.

Способи форматування задають структуру даних і відрізняються один від одного. Для того, щоб комп'ютери і програми могли читати і обробляти дані, структури файлів повинні відповідати певним правилам. Поширені формати на етапі додрукарської обробки видання: TIFF, EPS і PDF.

#### Растрові формати

---

- APNG — Animated PNG
- BMP
- ECW
- DRG — digital raster graphic
- GIF
- ICO (Windows Icon)
- ILBM
- JPEG
- JPEG 2000
- JPEG XR
- MNG
- PBM, PGM, PPM, PNM
- PCX
- PNG
- PSD
- TGA
- TIFF
- WMP
- XPM

#### Векторні формати

---

##### 2D

- SVG та SVGZ (Scalable Vector Graphics)
- EPS (Encapsulated PostScript)
- Метафайли Windows: WMF, EMF
- Файли CorelDraw: CDR, CMX
- AI (Adobe Illustrator)
- XAR

##### 3D

- COLLADA — формат, розроблений для обміну між 3D додатками
- SKP
- STL — A stereolithography format
- U3D — Universal 3D file format
- VRML — Virtual Reality Modeling Language
- X3D

- .3ds
- 3DXML

## Комплексні формати

---

- CDF
- DjVu
- EPS
- PDF
- PICT
- PS
- SWF
- XAML

### 4. Методи стискання та кодування графічних даних.

Растрові файли мають дуже великі розміри . Якщо знехтувати заголовками файла й іншими неграфічними даними, те його розмір пропорційний кількості пікселів у зображенні і кількості бітів, необхідних для представлення кожного піксела. Повнокольорове зображення розміром 1024x768 пікселів займає більш двох мегабайт пам'яті, а одна секунда відеофільма телевізійної якості в растровому виді вимагає біля тридцятьох мегабайт. Тому жорсткий диск можна заповнити миттєво. Навіть компакт-диск, що вміщає біля 700 мегабайт даних, не настільки великий, щоб помістити такий об'єм інформації.

Використовуючи метод, який називається стисненням зображень, можна різко зменшити в розмірі графічні файли. При стисненні графічної інформації використовуються спеціальні прийоми, що зменшують кількість байтів, необхідних для представлення зображення. Степінь стиснення залежить від методу стиснення і вмісту графічного файла . Як правило графічний файл стискується в п'ять і більш разів. Існують методи, що стискають ще сильніше, але з втратами якості . При відновленні зображення втрачається деяка частина колірної інформації. У підсумку, розпаковане зображення може стати злегка розмитим або знебарвленим.

Методи стиснення растрової інформації діляться на дві великі групи: стиснення з втратами і стиснення без втрат. Методи стиснення без втрат дають більш низький коефіцієнт стиснення, але зате зберігають точно значення пікселів вихідного зображення. Методи з втратами дають більш високі коефіцієнти стиснення, але не дозволяють відтворити початкове зображення з точністю до піксела. Для файлів, які формуються програмами автоматизованого проектування, дуже важливо зберегти всю інформацію, тому що втрата хоча б одного біта може змінити зміст усього файла. Зовсім інша справа з растровими даними. Людське око не сприймає всі відтінки кольору в звичайному растровому зображенні. Таким чином, деякі деталі можуть бути опущені без видимого порушення інформаційного змісту зображення.

Розглянемо два найбільш розповсюджені методи стиснення зображень. Спочатку познайомимося з одним із варіантів групового кодування (run-length encoding - RLE). Ідея методу полягає в тому, що послідовність значень, що повторюються, замінюється парою чисел: одне з них вказує на довжину групи (число повторень даного значення), а інше - на власне це значення. Це дуже загальний і дуже простий метод без втрат. Він використовується в багатьох популярних сьогодні форматах графічних файлів і, зокрема, у PCX і BMP. У його основі лежить той факт,

що багато зображень надлишкові, оскільки містять велику кількість суміжних пікселів одного кольору. Розглянемо, наприклад, як за допомогою групового кодування стискається зображення, у якому зустрічається підряд 100 пікселів із нульовим значенням. Ця послідовність із 100 нулів кодується парою чисел (100,0). Отже такий фрагмент картинки скоротиться в п'ятдесят разів.

Інший методом, яким користуються досить часто, - JPEG (метод, що стискує з утратами) одержав свою назву від аббревіатури об'єднаної групи експертів в області фотографії (Joint Photographic Expert Group - JPEG), що його і розробила. JPEG широко використовується при стисненні статичних зображень. Цей метод істотно складніший, чим RLE. Основна ідея методу перебуває в поділі інформації в зображенні за рівнем важливості, і потім відкиданні менше важливої її частини, зменшуючи тим самим загальний об'єм збережених даних. Це досягається перетворенням матриці кольірних значень у матрицю амплітуд, що відповідають визначеним частотам розкладання зображення. (Звукові коливання, наприклад, можна розкласти математичними методами на прості синусоїдальні гармоніки різних амплітуд і частот, що при додаванні відтворюють вихідний сигнал). Рядок або стовпець пікселів зображення теж можна представити амплітудами і частотами. Мова в даному випадку йде не про спектральний склад світла, а про форму представлення кривих, що утворюють графіки, якщо значення пікселів служать ординатами. Відзначимо, що формула перетворення матриці пікселів у матрицю амплітуд не проста. JPEG-стиснення відкидає частину високочастотних компонент зображення, залишаючи компоненти з низькими частотами. Людське око менше критичне до високочастотних варіацій кольору, оскільки загальний вид зображення визначається низькими частотами. Значення піксела, отримане при відновленні зображення, дещо відрізняється від вихідного значення, тому що частина інформації була загублена, хоча звичайно вони дуже близькі.

У методу JPEG є дуже цікава особливість: користувач може задавати коефіцієнт якості. Високий коефіцієнт якості дозволяє зберегти більше деталей, але при цьому зменшується ступінь стиснення. При низькому коефіцієнті якості ступінь стиснення збільшується, але зображення стає менше чітким.

Чим нижче коефіцієнт якості, тим більша кількість інформації відкидається.

Коли любий із методів (RLE або JPEG) застосовується до повнокольорового зображення, то червона, зелена і синя компоненти стискаються незалежно. Якщо в растровому зображенні використовується палітра або просто відтінки сірого, то значення пікселів можливо закодувати в один прохід.

### **Алгоритм групового кодування**

Починаючи з першого рядка, програма групового кодування переглядає значення пікселів зліва праворуч і шукає відрізки пікселів, що повторюються. Всякий раз, коли зустрічаються три або більше пікселів, що йдуть підряд, з однаковим значенням, програма заміняє їх парою чисел: перше число вказує на довжину відрізка, друге - на значення пікселів. Число, що визначає довжину відрізка, називають міткою відрізка.

Щоб ідентифікувати серії значень пікселів, що не повторюються, програма також уставляє мітки, що вказують на кількість таких значень у серії. Зарезервований біт необхідний для того, щоб можна було відрізнити мітку відрізка від мітки серії значень, що не повторюються. Наприклад, у 8-ми бітах можна задати послідовності довжиною до 127 пікселів; восьмий біт у кожній мітці може відрізнити відрізок від



серії пікселів, що не повторюються. Точно так само обробляється кожний рядок пікселів і відрізки однакових значень пікселів стискаються у всьому зображенні.

Графічна програма декодує зображення, зчитуюючи стиснутий файл і відновлює відрізки повторюваних значень пікселів. Зауважимо, що відновлене зображення цілком збігається з оригіналом.

### **Алгоритм JPEG**

Насамперед програма поділяє зображення на блоки - матриці розміром 8x8 пікселів. При використанні методу JPEG час, що затрачається на стиснення зображення, пропорційний квадрату числа пікселів у блоці. Обробка декількох блоків меншого розміру робиться значно швидше, чим обробка всього зображення цілком.

До значень пікселів застосовується формула, названа дискретним косинусоїдальним перетворенням (Discrete Cosine Transform - DCT). DCT переводить матрицю значень пікселів 8x8 у матрицю значень амплітуд тієї ж розмірності, що відповідає визначеним частотам синусоїдальних коливань. Лівий верхній кут матриці відповідає низьким частотам, а правий нижній - високим.

Коефіцієнт якості, введений користувачем, використовується в простій формулі, що генерує значення елементів іншої матриці 8x8, яка називається матрицею квантування. Чим нижче коефіцієнт якості, тим більші значення будуть мати елементи матриці.

Кожне значення в матриці, яка була сформована після DCT-перетворення, ділиться на відповідне значення з матриці квантування, потім округляється до найближчого цілого числа. Оскільки великі числа знаходяться в правій нижній половині матриці квантування, то основна частина високочастотної інформації зображення буде відкинута. Тому нижня права частина матриці пікселів буде перебувати в основному з нулів.

Далі програма зчитує елементи матриці і кодує їх послідовно методами без втрат. Зауважимо, що стиснення істотно залежить від нулів у правій нижній половині матриці. Чим нижче коефіцієнт якості, тим більше нулів у матриці і, відповідно, тим вище ступінь стиснення.

Декодування JPEG-зображення починається з кроку зворотного кодування без втрат, у результаті чого відновлюється матриця квантування пікселів.

Значення з матриці пікселів перемножується на значення з матриці квантування, щоб відновити, наскільки це можливо, матрицю, що була обчислена на кроку застосування DCT. На етапі квантування була загублена деяка частина інформації, тому числа в матриці будуть близькі до початкових, але не буде абсолютного збігу.

Зворотна до DCT формула (IDCT) застосовується до матриці для відновлення значень пікселів вихідного зображення. Ще разом відзначимо, що отримані кольори не будуть цілком відповідати початковим через втрату інформації на кроку квантування. Відновлене зображення, при порівнянні з оригіналом, буде виглядати декілька розмитим і знебарвленим.

### **Алгоритм Хаффмана**

Один з класичних алгоритмів. Використовує тільки частоту появи однакових байт в зображенні. Порівнює символів вхідного потоку, які зустрічаються більше число разів, ланцюжок біт меншої довжини. І навпаки - зустрічаються рідко - ланцюжок більшої довжини. Для збору статистики вимагає двох проходів по зображенню. Коефіцієнти стиснення: 1 / 8, 2 / 3, 1. Вимагає запису у файл таблиці

відповідності кодуються символів і кодують ланцюжків. На практиці використовуються його різновиди. Так, в деяких випадках резонно або використовувати постійну таблицю, або будувати її "адаптивно", тобто в процесі архівації / розархівації. Ці прийоми позбавляють від двох проходів по зображенню і необхідності зберігання таблиці разом з файлом. Кодування з фіксованою таблицею застосовується як до останнього етапу архівації в JPEG.

Близька модифікація алгоритму використовується при стисненні чорно-білих зображень. Послідовності посліпль йдуть чорних і білих крапок замінюються числом, рівним їх кількості з ознакою кольору. А цей ряд вже, у свою чергу, стискується по Хаффману з фіксованою таблицею. Алгоритм реалізований у форматі TIFF

### **JBIG**

Алгоритм розроблений групою експертів ISO (-Joint Bi level Experts Group) спеціально для стиснення однобітних чорно-білих зображень. Наприклад, факсів або відсканованих документів. У принципі може застосовуватися і до 2-х, і до 4-х двійкового картинок. При цьому алгоритм розбиває їх на окремі бітові площині. JBIG дозволяє управляти такими параметрами, як порядок розбиття зображення на бітові площині, ширина смуг в зображенні, рівні масштабування. Остання можливість дозволяє легко орієнтуватися в базі великих за розмірами зображень, переглядаючи спочатку їх зменшені копії. Настроюючи ці параметри, можна використовувати цікавий ефект при отриманні зображення по мережі або з будь-якого іншому каналу, пропускна здатність якого мала в порівнянні з можливостями процесора. Розпаковуватися зображення на екрані буде поступово, як би повільно "проявляючись". При цьому людина починає аналізувати зображення задовго до кінця процесу розархівації.

Алгоритм побудований на базі Q-кодувальника, патент на який володіє IBM. Q-кодер також, як і алгоритм Хаффмана, використовує для частіше з'являються символів короткі ланцюжки, а для рідше з'являються довгі. Однак, на відміну від нього, в алгоритмі використовуються і послідовності символів. Характерною особливістю JBIG є різке зниження ступеня стиснення при підвищенні рівня шумів вхідний картинки.

### **Фрактальний стиск**

Ця група алгоритмів, мабуть, є найбільш перспективною і розвивається зараз найбільш бурхливо. Перші практичні результати були отримані зовсім недавно - у 1992 році - і виробили приголомшливе враження. Коефіцієнт стискування у фрактальних алгоритмів варіюється в межах 2-2000. Причому великі коефіцієнти досягаються на реальних зображеннях, що, взагалі кажучи, нетипово для попередніх алгоритмів. Крім того, при розархівації зображення можна масштабувати. Унікальна особливість цього алгоритму полягає в тому, що збільшене зображення не дробиться на квадрати. Під фрактального стиснення використовується принципово нова ідея - не близькість квітів у локальній області, а подібність різних за розміром областей зображення. Це, безумовно, найбільш прогресивний підхід на сьогоднішній день. Алгоритм орієнтований на повнокольорові зображення і зображення в градаціях сірого кольору.

Його особливістю є потреба в колосальних обчислювальних потужностях за архівації. При цьому розпаковування вимагає менше обчислень, ніж у JPEG.

## **V. Закріплення нових знань.**

Бесіда за запитаннями:

- Які кольорові моделі Ви запам'ятали?
- Охарактеризуйте колірну модель **СМΥΚ**
- Які графічні формати Вам відомі?
- Охарактеризуйте колірну модель **HSB**
- Дайте визначення кольорокорекції

**Домашнє завдання:**

1) Законспекуйте матеріал уроку

2) **підручник:** В.В. Вдовченко «Основи дизайну», 10 клас, Профільне навчання, К. Педагогічна думка, 2010- §15

**3) Зворотній зв'язок за ел. поштою:** [2573562@ukr.net](mailto:2573562@ukr.net)