



**ЛЬВІВСЬКА ОБЛАСНА ДЕРЖАВНА АДМІНІСТРАЦІЯ
ДЕПАРТАМЕНТ ОСВІТИ І НАУКИ**

79008, м. Львів, вул. Просвіти, 4а; тел./факс 261-63-54, e-mail donloda@ukr.net, код ЄДРПОУ 38557639

**Керівникам органів управління
освітою**

***Щодо проведення обласного конкурсу
«Закодуй майбутнє»***

З метою популяризації ІТ-освіти та розвитку навичок програмування у дітей шкільного віку, а також з метою підтримки талановитої молоді Львівської області оголошується конкурс від волонтерської мережі Code Club Ua «Закодуй майбутнє».

Організатори конкурсу: Code Club Ua за підтримки департаменту освіти і науки Львівської обласної державної адміністрації

Умови конкурсу:

1. Зареєструвати заклад загальної середньої освіти на участь в Code Club Ua <https://cutt.ly/YjaimaF>
2. У конкурсі беруть участь команди у складі трьох учнів та педагога-наставника. Один заклад загальної середньої освіти може представити проекти декількох команд.
3. Педагог-наставник реєструє команду та учасників.
4. Команда учасників має подати проєкт у форматі гри, анімації або додатку в одній із категорій:
 - I. Scratch – проєкт (учасники 8-10 років)
 - II. HTML – власний креативний сайт (10-13 років)
 - III. Python – проєкт (13-16 років)

Конкурс проводиться у 4 етапи:

I етап - реєстрація команд з 02 березня по 15 березня 2021 року.

<https://cutt.ly/LkYUjDh>

II етап – робота над проєктом з 15 березня по 30 квітня

III етап - відбірковий (онлайн). Презентація проєктів на платформі ZOOM та визначення переможців 15 травня 2021 року.

IV етап – нагородження переможців на одній з локацій Code Club Ua 29 травня 2021 року.

Команди-фіналісти будуть запрошені до Львова для вручення дипломів, призів та цінних подарунків у кожній з представлених категорій. Також на фіналістів очікує екскурсія в ІТ компанію та зустріч з представниками ІТ кластеру.

Організатори забезпечують учасників харчуванням та кава-брейками, витрати на переїзд до Львова оплачуються самостійно.

Критерії оцінювання:

- ☐ коректність використання блоків алгоритму та розуміння принципів їх роботи;
- ☐ логіка алгоритму;
- ☐ наявність помилок;
- ☐ лаконічність алгоритму та різноманітність команд;
- ☐ багаторівневність проєкту;
- ☐ візуальне оформлення;
- ☐ зручність користування, інтерфейс.