

Навчальна програма з позашкільної освіти
науково-технічного напрямку
інформаційно-технічного профілю
«СУЧАСНІ ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ»
1 рік навчання

Чебанюк О. В., керівник гуртка лабораторії комп'ютерних технологій
відділу науково-технічної творчості
Київського Палацу дітей та юнацтва;

Белоглазова Н. В., начальник відділу змісту позашкільної освіти
відділення змісту позашкільної освіти та виховної роботи
Інституту інноваційних технологій і змісту освіти

ПОЯСНЮВАЛЬНА ЗАПИСКА

Актуальність навчальної програми пов'язана з широким проникненням інформаційних технологій практично в усі сфери сучасного життя та зростанням попиту на ІТ-фахівців. Учнівська молодь активно цікавиться вивченням дуже потужної та водночас простої у використанні мови програмування Microsoft C#, яку високо цінують сучасні професіонали галузі інформаційних технологій. Зазначена мова програмування є вдалим вибором для тих, хто власне тільки починає вивчати програмування.

Особливістю навчальної програми є досить докладний опис мови C#, наявні способи навчання дітей щодо вирішування алгоритмічних завдань, з використанням мови програмування C#, та застосування її для розвитку логічного мислення, прищеплення інтересу до розв'язання алгоритмічних задач та подальшого вивчення сучасних мов і технологій програмування. Програма містить інформацію про вбудовані типи даних, керівні структури мов, процедури, рекурсії, рядки, масиви. Акцент зроблено на класах, структурах, інтерфейсах, делегатах і подіях, на вивченні взаємозв'язку між класами. Описано середовище розробки Visual Studio. Net, класи бібліотеки FCL, організація інтерфейсу Windows-проектів.

Організація навчання здійснюється за лінійним принципом, коли матеріал вивчають систематично і послідовно, з поступовим ускладненням. Особливу увагу зосереджено на розвитку дослідницьких умінь дітей, пошуку власних способів виконання завдань.

При підготовці програми були враховані основні положення Законів України «Про основні засади розвитку інформаційного суспільства в Україні на 2007–2015 роки» та «Про національну програму інформатизації», Положення про порядок організації індивідуальної та групової роботи в

позашкільних навчальних закладах (у редакції наказу Міністерства освіти і науки України від 10.12.2008 р. № 1123) та інших нормативних документів.

Головна мета програми — формування вмінь і навичок, необхідних для розробки прикладних програм мовою C#.

Основні завдання програми:

- засвоєння об'єктно-орієнтованих мов програмування Microsoft C#, механізмів та технологій, що базуються на використанні та застосуванні класів.
- формування вміння самостійної розробки програми (постановка завдання, створення математичної та логічної моделі, проектування алгоритму методом покрокової деталізації, створення та зневадження програми).

Навчання групи 12–15 вихованців передбачене для дітей віком 15–17 років протягом 1 року (основний рівень - 216 годин, 6 годин на тиждень).

Основною формою проведення занять є групова. Практична робота виконується вихованцями як індивідуально, так і у групах, відповідно до обраної керівником гуртка методики, і передбачає виконання конкретного однакового для всіх завдання.

Навчально-тематичний план програми є орієнтовним. Розділи розподілені на теми із зазначенням обсягу годин, необхідних для виконання кожного виду навчальних занять та самостійної роботи. Керівник гуртка може змінювати кількість теоретичних і практичних занять (залежно від того, як швидко та якісно вихованці засвоюють теоретичні знання та набувають практичних навичок), враховуючи рівень інтелектуального розвитку гуртківців, математичної підготовки. Різниця розподілу навчальних годин вказується в навчально-тематичному плані. Учні обирають теми творчих проектів самостійно за особистим уподобанням та інтересом, але відповідно до навчальної тематики, рівня складності, віку, а також за порадою керівника гуртка. Є можливість виконувати індивідуальні, парні та колективні проекти.

Основний принцип роботи педагога полягає в доступному викладенні учням матеріалу навчального курсу. Тому глибина вивчення тем залежить від рівня математичної підготовки учнів. Психологічний клімат у групі формується на основі творчості та взаємоповаги між учнями та викладачем.

Знання та вміння контролюються поточним оцінюванням виконання практичних і самостійних завдань, запланованих відповідно до програми, а також участю учнів у конкурсах, змаганнях і олімпіадах. Основними способами контролю ефективності навчання є тестування, співбесіда, система заліків, контрольні завдання, захист власних проектів, а також іспит.

При формуванні групи враховуються вікові особливості дітей та зміст програми.

До навчання в гуртку (секції, творчому об'єднанні тощо) зараховуються діти за результатами тестування, оскільки для опанування програми «Системне програмування» потрібно мати певний рівень знань і вмінь користувача-початківця.

Програмне забезпечення, необхідне для вивчення запланованого навчального матеріалу:

- операційна система Windows XP, Windows 7 або Linux;
- інтегроване середовище розробки (IDE) програм мовою C++ (Microsoft Visual Studio 10, Microsoft Visual Studio Express, або Borland C++ Builder);
- програми архівування даних для роботи з архівними файлами;
- набір програм для роботи в глобальній мережі Інтернет для перегляду web-сторінок, роботи з електронною поштою та телеконференціями, файловими архівами, пошуковими системами тощо.

Для якісного навчання 12–15 вихованців слід забезпечити кожного індивідуальним робочим місцем.

Обов'язковою умовою є належний технічний стан комп'ютерів (достатній обсяг оперативної пам'яті та жорсткого диску, наявність DVD-ROM, рознімів для USB флеш-пам'яті; доступ до мережі Інтернет; комп'ютери повинні бути об'єднані в локальну мережу) та наявність необхідного програмного забезпечення, а також периферійних пристроїв. Навчальний процес має бути забезпечений наочним і дидактичним матеріалом (спеціальною методичною літературою; Інтернет-ресурсами, освітньо-інформаційними ресурсами).

Основний рівень, один рік навчання НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

Розділ, тема	Кількість годин		
	теоретичних	практичних	усього
Вступ	3	–	3
Розділ 1. Мова C #	27	33	60
1.1. Система типів мови C#	9	9	18
1.2. Змінні та вирази. Регулярні вирази	9	12	21
1.3. Оператори мови C#	9	12	21
Розділ 2. Об'єктно-орієнтована технологія програмування	63	87	150

2.1. Класи. Наслідування. Поліморфізм	21	27	48
2.2. Інтерфейси. Множинне наслідування	15	18	33
2.3. Функціональний тип у C#. Делегати	15	18	33
2.4. Події. Обробка виняткових ситуацій	12	15	27
2.5. Робота над фінальним проектом	–	9	9
Підсумок	3	–	3
Разом:	96	120	216

ЗМІСТ ПРОГРАМИ

Вступ (3 год.)

Теоретична частина. Сучасні професії в галузі інформаційних комп'ютерних технологій (ІКТ). Вимоги до сучасних ІТ-фахівців. Вступний інструктаж з безпеки життєдіяльності вихованців. Організаційні питання.

Розділ 1. Мова C# (60 год.)

1.1. Система типів мови C# (18 год.)

Теоретична частина. Створення мови. Особливості мови. Рішення, проекти, простір імен. Консольні та Windows-програми C#, побудовані типово.

Загальна характеристика. Система типів. Типи-значення і посилальні типи. Вбудовані типи. Порівняння з типами C++. Перетворення змінних на об'єкти та vice versa. Операції «упакувати» та «розпакувати».

Практична частина. Створення найпростішої консольної прикладної програми з використанням простих типів даних: int, double, string.

1.2. Змінні та вирази. Регулярні вирази (21 год.)

Теоретична частина. Оголошення змінних. Синтаксис оголошення. Ініціалізація. Час життя та ділянка видимості. Де оголошуються змінні? Локальні та глобальні змінні. Константи.

Простір Regular Expressions і його класи. Регулярні вирази. Теорія регулярних виразів. Практика застосування регулярних виразів. Розбір текстів і пошук за зразком. Властивості та методи класу Regex та інших класів, пов'язаних з регулярними виразами. Приклади застосування регулярних виразів.

Практична частина. Створення програми з використанням регулярних виразів. Дослідження властивостей та методів класу Regex.

1.3. Оператори мови C# (21 год.)

Теоретична частина. Оператори мови C#. Оператор присвоювання. Складений оператор. Порожній оператор. Оператори вибору. Оператор if. Оператор switch. Оператори переходу. Оператор goto. Оператори break, continue. Оператори циклу. Оператор for. Цикли while. Цикл foreach.

Загальна характеристика масивів. Порівняння з масивами C++. Чому масиви C# кращі, ніж масиви C++? Види масивів: одномірні, багатомірні та порізані. Динамічні масиви.

Практична частина. Обробка одновимірних та багатовимірних масивів з використанням операторів for, foreach, while.

Розділ 2. Об'єктно-орієнтована технологія програмування (150 год.)

2.1. Класи. Наслідування. Поліморфізм (48 год.)

Теоретична частина. Дві ролі класу в об'єктно-орієнтованому програмуванні. Синтаксис опису класу. Поля та методи класу. Конструктори та деструктори. Статичні поля та методи. Статичні конструктори. Поля тільки для читання. Закриті поля. Стратегії доступу до полів класу. Процедури властивостей. Індиксатори. Приклади.

Взаємозв'язок між класами. Взаємозв'язок «клієнти — постачальники». Взаємозв'язок наслідування. Одиначне наслідування. «Батьки» та «спадкоємці». «Предки» та «нащадки». Що успадковують нащадки та що можуть змінити? Однобічне присвоювання. Контроль типів і зв'язування. Поліморфізм. Проектування класів. Абстрактні класи. Класи поведінки.

Практична частина. Написання програм з використанням створених класів, що реалізують механізм наслідування. Створення діаграми класів.

2.2. Інтерфейси. Множинне наслідування (33 год.)

Теоретична частина. Інтерфейси як окремий випадок класу. Множинне наслідування. Проблеми. Множинне наслідування інтерфейсів. Вбудовані інтерфейси. Інтерфейси IComparable, ICloneable, ISerializable. Поверхневе та глибоке клонування й серіалізація. Збереження й обмін даними.

Практична частина. Створення програми сортування масивів складних об'єктів з перевизначенням методу інтерфейсу IComparable. Дослідження методів ICloneable, ISerializable.

2.3. Функціональний тип у C#. Делегати (33 год.)

Теоретична частина. Нове слово для старого поняття. Функціональний тип. Функції вищих порядків. Два способи взаємодії частин під час побудови

складних систем. Функції зворотного виклику. Наслідування та функціональні типи. Порівняння двох підходів. Клас Delegate. Методи та властивості класу. Операції над делегатами. Комбінування делегатів. Список викликів.

Практична частина. Розробка програми з використанням методів класу Delegate. Обчислення інтеграла та сортування.

2.4. Події. Обробка виняткових ситуацій (27 год.)

Теоретична частина. Класи з подіями. Загальна характеристика. Клас Sender і класи Receivers. Клас Sender. Як оголошуються події? Делегати та події Net Framework. Класи з допустимими подіями. Net Framework. Клас EventArgs і його нащадки. Вхідні та вихідні аргументи події. Клас Receiver. Обробник події. Вбудовування об'єкта Sender. Зв'язок обробника з подією. Відключення обробника. Взаємодія об'єктів sender і receiver. Події — поля чи процедури-властивості? Динамічне пов'язання подій з їх обробниками.

Коректність і стійкість. Специфікація системи. Коректність і стійкість програмних систем. Виняткові ситуації. Обробка виняткових ситуацій. Зневадження. Створення надійного коду. Налаштування й інструментарій Visual Studio .Net.

Практична частина. Створення програми, що використовує обробники подій. Створення надійної прикладної програми за рахунок обробки виняткових ситуацій.

2.5. Робота над фінальним проектом (9 год.)

Практична частина. Використовуючи набуті навички та теоретичні знання, учні розробляють власний проект за запропонованими темами: «Власна прикладна система».

Розробка об'єктно-орієнтованої прикладної програми.

Підсумок (3 год.)

Теоретична частина. Підбиття підсумків роботи учнів за навчальний період.

ПРОГНОЗОВАНІ РЕЗУЛЬТАТИ

Вихованці мають знати і розуміти:

- призначення та основні функції інтегрованого середовища microsoft visual studio 2010;
- правила роботи з інтегрованим середовищем microsoft visual studio 2010;

- операції, які можна виконувати з інтегрованим середовищем microsoft visual studio 2010;
- правила розробки об'єктів;
- принципи побудови ієрархії об'єктів;
- технологію створення класів;
- правила розробки графічних та мультимедійних прикладних програм;
- тонкощі програмування під windows та використання api функції;
- основні відомості про реляційні бази даних;
- компоненти для роботи з базами даних;
- принципи технології client/server.

Вихованці мають вміти і застосовувати:

- працювати з object inspector;
- редагувати object event;
- використовувати властивості та події mplayer;
- розробляти структуру одноярусної бази даних;
- виконувати запити до бази даних;
- змінювати конфігурацію панелі компонентів;
- створювати об'єкт відповідно до правил структуризації програм;
- розробляти одноярусну базу даних;
- створювати власні класи.

Вихованці мають набути досвід:

- складання програм різного рівня складності.
- налагодження програм (td та cru) з використанням відповідного інструментарію;
- розробки одноярусної бази даних;
- використання об'єкта gdi для роботи з графікою

ЛІТЕРАТУРА

1. Бен Ватсон. С# 4.0 на прикладах. — М.: «Вільямс», 2010. — 610 с.
2. Герберт Шилдт. С++: базовий курс (C++ from the Ground Up). — 4-е изд. — М.: Вільямс, 2008 — 624 с.
3. Закон України «Про освіту» // Освіта України. — 1996. — № 123.
4. Закон України «Про позашкільну освіту» // Урядовий кур'єр. — 2000. — № 25.
5. Лехан С. А. Інформатика. Мова програмування С++. Спецкурс. — Шепетівка: «Аспект», 2007. — 159 с.

6. М. Дрейер. С# для школьников: Учебное пособие / Перевод с англ. под ред. В. Биллига — М.: Интернет-университет информационных технологий; БИНОМ. Лаборатория знаний, 2009. — 128 с.
7. Мартынов Н. Н. С# для начинающих. — М.: КУДИЦ-ПРЕСС, 2007. — 272 с.
8. Мэтью Мак-Дональд. Silverlight 3 с примерами на С# для профессионалов. — М.: «Вильямс», 2010. — 560 с.
9. Н. Культин. С/С++ в задачах и примерах. — СПб.: «БХВ-Петербург», 2009. — 368 с.
10. Положення про позашкільний навчально-виховний заклад // Інф. зб. Міносвіти України. — 1994. — № 9.
11. Програми з позашкільної освіти: науково-технічний напрям / Биковський Т. В., Вихренко Т. О. та ін. — К., 2012.
12. Рейтинг языков программирования — июнь 2011. [Электронный ресурс] - Режим доступа:
http://studlab.com/news/rejting_jazykov_programmirovanija/2011-06-09-125.
13. Т. Ковалюк. Основи програмування. — К.: «Видавнича група ВНУ», 2005. — 384 с.
14. Т. Павловская. С/С++. Программирование на языке высокого уровня. — СПб.: «Питер», 2007. — 461 с.