

ТЕОРЕТИЧНІ ОСНОВИ ПОБУДОВИ ПРОЕКЦІЙНИХ МОДЕЛЕЙ ТРИВИМІРНОГО ЕВКЛІДОВОГО ПРОСТОРУ НА ПЛОЩИНІ

Постановка проблеми. Створити міжпредметний навчальний курс з алгебри, геометрії і програмування мовою Python 2.7. Призначення програми – побудова перспективних проекцій довільної фігури заданої параметрично, з можливістю задання користувачем своїх формул. Програма реалізовує побудову проекцій шляхом матричних операцій над однорідними координатами.

Актуальність проблеми дослідження. Актуальність проблеми міжпредметних зв'язків у сучасних умовах посилюється, оскільки міжпредметні зв'язки – найважливіший фактор оптимізації процесу навчання, підвищення його результативності, зменшення перенавантаження вчителів і студентів.

Аналіз останніх досліджень і публікацій. Була розглянута велика кількість англomовних матеріалів по можливостях Python 2.7 і подійній моделі поведінки в PyQt4. Також були проаналізовані останні публікації по використанню матричного методу при побудові графічних зображень. Однак є велика прогалина в прикладах документації PyQt4 по створенні елементів контекстного меню. Оскільки об'єм статті обмежений і в початківців виникає найбільше проблем саме з ним я опишу тільки цю частину роботи.

Виклад основного матеріалу дослідження.

Для створення інтерфейсу користувача програмного продукту була обрана бібліотека PyQt4 (через відсутність в Python вбудованих функцій). Саме ця бібліотека була вибрана через її Open Source статус. Створення інтерфейсу відбувається через редактор Qt Designer. Як інтегроване середовище розробки рекомендується використовувати PyCharm Professional Edition котра є безкоштовною для освітніх установ.

PyQt використовує подійну модель поведінки. Це значить що ми описуємо програму як список реакцій на певні події "events". Qt Designer є середовищем візуального програмування в котрому ми просто поміщуємо компоненти графічного інтерфейсу на вікно програми. Для того щоб зв'язати натиск на кнопку з виконанням певної Python функції нам також необхідно описати "signal" і "slot". Кожен PyQt віджет породжений від класу QObject і розроблений таким чином щоб посилати signal як реакцію на одну або більше подій. Сигнал сам по собі не виконує жодної функції, натомість він з'єднаний з "slot". Slot – це будь-яка Python функція. Нехай у нас є елемент контекстного меню котрий відповідає за ввімкнення вимкнення анімації з назвою actionAnim. Для того щоб здійснити з'єднання "slot" і "signal" необхідно виконати наступну команду self.connect(self.actionAnim, QtCore.SIGNAL("triggered(bool)"), self.view.animate). котра зв'яже сигнал від цього віджета з функцією animate. Іншими словами при натисканні на кнопку її стан (увімкнена\вимкнена) буде сконвертований в булеву змінну котра в подальшому буде передана як аргумент в функцію animate. В свою чергу функцію буде необхідно описати наступним чином:

```
@QtCore.pyqtSignature("bool")
```

```
def animate(self, state):
    if state:
        self.time1 = self.startTimer(50)
        self.time2 = self.startTimer(100)
        self.time3 = None
        self.i = 0
        self.b = 0
        self.animationsIsEnable = state
```

Стандартний синтаксис Python за винятком додавання анотації PyQt перед описом імені функції. Створення контекстного меню виконується лише кодом, ось витяг з вихідного коду готового проекту як його додавати.

```
self.menuIn.addSeparator()
```

```
self.invisible = self.menuIn.addAction(self.trUtf8("тільки  
каркас"))
self.invisible.setCheckable(True)
self.invisible.setChecked(False)
self.connect(self.invisible, QtCore.SIGNAL("triggered( bool )"),
self.invisibility)
```

```
self.paintShadow = self.menuIn.addAction(self.trUtf8("тінь"))
self.paintShadow.setCheckable(True)
```

```
self.paintShadow.setChecked(True)
self.connect(self.paintShadow, QtCore.SIGNAL("triggered( bool
)"), self.shadow)
```

```
self.perspectiva =
self.menuIn.addAction(self.trUtf8("перспектива"))
self.perspectiva.setCheckable(True)
self.perspectiva.setChecked(False)
self.connect(self.perspectiva, QtCore.SIGNAL("triggered( bool
)"), self.itemsPro)
```

```
self.menuIn.addSeparator()
self.ac = self.menuIn.addAction(self.trUtf8("колір осей"))
self.connect(self.ac, QtCore.SIGNAL("triggered( )"),
self.carSysChangeColor)
```

ЛІТЕРАТУРА

1. Тадеєв В. О. Геометрія. Фігури обертання. Векторно-координатний метод: Дворівневий підручник для 11 класу загальноосвітніх навчальних закладів/ За ред. М. Я. Ядренка. – Тернопіль: навчальна книга – Богдан, 2004. – 480 с.
2. Інженерна та комп'ютерна графіка: Підручник / В. Є. Михайленко, В. М. Найдис, А.М. Підкоритов, І.А. Скидан; За ред. В.Є. Михайленка. – 2-ге вид., перероб. –К.: Вища шк., 2001. –350 с.:іл.
3. Маценко В.Г. Комп'ютерна графіка: Навчальний посібник. – Чернівці: Рута, 2009 – 343 с.
4. Ламот Андре Программирование трехмерных игр для Windows. Советы профессионала по трехмерной графике и растеризации.:Пер. с англ. — М. : Издательский дом "Вильямс", 2004. — 1424 с.: ил.
5. Головчук А. Ф., Кепко О. І., Чумак Н. М. Інженерна та комп'ютерна графіка: Навч. посіб. — К.: Центр учбової літератури, 2010. — 160 с.
6. Казанцев А.В. Основи комп'ютерної графіки. Тексти спеціального курсу лекцій. Частина 1. Математичний апарат комп'ютерної графіки. 2001 рік.
7. Роджерс Д., Адамс Дж. Математические основы машинной графики. М.: Мир, 2001. 604 с.
8. <https://www.python.org/>
9. <https://www.lfd.uci.edu/~gohlke/pythonlibs/#pyqt4>
10. <https://www.qt.io/download>

Поступмент М.

Науковий керівник – доц. Грод І.М.

ПОБУДОВА МОДЕЛІ АRІМА ДЛЯ ПРОГНОЗУВАННЯ ДИНАМІКИ ЧИСЕЛЬНОСТІ ПОПУЛЯЦІЇ

Актуальність роботи. У програмах екологічного моніторингу важливе місце відводиться розробці методів моделювання динаміки популяцій, а також вивченню можливостей оцінювати стан екосистем, угруповань і популяцій за особливостями варіювання чисельності (Geisser, Sauer, 1990). Аналіз довготривалих рядів спостережень є одним з основних завдань екологічного моніторингу [7]. Найбільш доступною інтегральною характеристикою популяцій є чисельність, з якою тісно пов'язано багато інших параметрів. Тому традиційно в теоретичній і практичній екології питанням вивчення динаміки чисельності приділяється першорядне значення [3]. Проте багато аспектів оцінки та аналізу чисельності популяцій дотепер залишаються дискусійними (Hurvich, Tsai, 1989; James, McCulloch, 1994; Kendall, Ord, 1990).

Отже, саме тому за умови постійного моніторингу за станом розвитку і динамікою змін та контролю за нормою вилучення особин популяції, а також за умови правильно здійсненого прогнозу популяція може існувати необмежено довгий час і зберігати свою продуктивність.

Метою роботи є розробка модифікованої моделі і відповідного їй методу прогнозування, що відноситься до класу авторегресійних моделей і повністю відповідає параметрам прогнозування чисельності окремих популяцій і їх співіснування.

Методи дослідження. При вирішенні поставлених завдань в роботі використані методи математичного моделювання, аналізу часових рядів, регресійного аналізу, методи алгоритмізації і програмування. Розглянемо модель ARIMA, врахувавши при цьому особливості заданої предметної галузі.

Постановка задачі. Нехай у деякому середовищі існує деяка популяція. Обмежень щодо площі розташування особин накладати не будемо.