

1. Зачем нужен OpenGL. [Електронний документ] - URL:<https://pozitive.org/windows/stati/opengl.html#novosib>.
2. Что такое OpenGL? [Електронний документ] - URL:<https://3dnews.ru/169184>.
3. Преимущества OpenGL. [Електронний документ] - URL:https://studbooks.net/2080851/informatika/preimuschestva_opengl.
4. В чем недостатки технологии OpenGL? [Електронний документ] - URL:<https://otvet.mail.ru/question/21388617>
5. Companies Currently Using Open Graphics Library. [Електронний документ] - URL:<https://discovery.hgdata.com/product/open-graphics-library-opengl>.

Ключові слова: бібліотека OpenGL, технологія обробки графіки, кроссплатформеність, розробка ігор.

Ключевые слова: библиотека OpenGL, технология обработки графики, кроссплатформеность, разработка игр.

Keywords: OpenGL library, graphics technology, cross-platform, game development.

Науковий керівник: к.т.н., доц. Задерейко О.В.

Сорокін Владислав Сергійович

*Національний університет «Одеська юридична академія»,
студент 2-го курсу факультету кібербезпеки та інформаційних
технологій*

МОВИ ПРОГРАМУВАННЯ ДЛЯ РОБОТОТЕХНІКИ

Сучасний етап розвитку науки й техніки характеризується зростанням популярності робототехніки та розширенням сфери використання роботів.

Загальні тенденції впровадження робототехніки в освіту змінились протягом останніх років після появи доступних компонентів для навчання робототехніки, що уможливило широке розповсюдження власноруч сконструйованих роботів.

За даними ResearchAndMarkets, очікується, що ринок промислової робототехніки зросте на 48 166,9 мільйона доларів США до 2025 року з 16 847,6 мільйонів доларів США у 2017 році [1].

Програмування роботів – це мистецтво навчання робота в режимі онлайн або офлайн для виконання певних завдань з необхідною точністю, швидкістю та повторюваністю. Програмування код має бути зрозумілим для робота, який по суті є програмованим багатофункціональним маніпулятором. Наприклад, промисловий робот призначений для переміщення матеріалів, деталей, інструментів або інших спеціалізованих пристроїв для виконання різноманітних завдань в межах заводського цеху.

Програмування промислового робота – це не лише кодування інструкцій за допомогою низькорівневої мови програмування. Оскільки технології, які лежать в основі робототехніки, продовжують розвиватися, з'явилися нові методи програмування, призначені для полегшення роботи користувачів.

З моменту виникнення програмованих обчислювальних систем засоби розробки керувального програмного забезпечення постійно вдосконалюються. Вони еволюціонували від машинних кодів та асемблера до мов програмування високого рівня та об'єктно орієнтованого програмування.

Комунікація між роботом та людиною відбувається за такою схемою [2]:

1. людина пише програму мовою програмування;
2. написаний код програми проходить через компілятори, інтерпретатори або транслятори і перетворюється на машинний код або байт-код для виконання віртуальною машиною, зрозумілий роботу;
3. машинний код потрапляє до «мозку» робота, роль якого може відіграти схема, чіп або мікроконтролер, де і виконується.

Для виконання певної дії роботи програмуються або за допомогою програмування в автономному режимі, або за допомогою наведення. Роботи отримують інструкції у вигляді комп'ютерних команд, і це називається

автономним програмуванням на рівні маніпулятора. Більшість промислових роботів запрограмовані шляхом наведення робота від точки до точки через фази операції, при цьому кожна точка зберігається в системі керування робота [1].

Використання офлайнового програмування передбачає мови високого рівня, в яких роботизовані дії визначаються завданнями або цілями.

Програмісти-роботи повинні володіти знаннями про різні типи мов програмування, оскільки перехід від комп'ютерів до роботів не є плавним переходом, як можуть думати багато розробників/програмістів.

Охарактеризуємо різні типи мов програмування роботів.

C/C++ – найпопулярніші мови програмування в робототехніці, які дозволяють високопродуктивно взаємодіяти з апаратним забезпеченням у реальному часі. Наприклад, мікроконтролер Arduino використовує C-подібну мову програмування. Завдяки численним інструментам, бібліотекам і функціям, C/C++ є лідером серед засобів програмування серед «інженерів-робототехніків». Саме C/C++ використовують, коли треба забезпечити максимальну продуктивність робота, тобто високу якість роботи на швидкій швидкості.

Python – це мова програмування високого рівня, яка відіграє ключову роль у створенні та тестуванні роботів. Ця мова програмування дозволяє написати сценарій, який обчислюватиме, записуватиме та моделюватиме всю програму робота замість того, щоб вручну навчати робота кожному оператору. Це допомагає швидко протестувати та візуалізувати рішення під час моделювання, а також уточнити програму та її логіку. Саме Python використовується з Raspberry Pi. Python є популярною мовою, завдяки її використанню в машинному навчанні, а також тому, що її можна використовувати для розробки пакетів роботизованих систем ROS (Robot Operating System). Python рекомендується для новачків, які починають працювати в робототехніці та програмуванні.

JAVA пропонує набір API, які створені відповідно до потреб сфери робототехніки. Розпізнавачі команд і керування, системи диктування та синтезатори мовлення можуть бути створені за допомогою Java Speech API, а Java Media Framework може використовуватися для отримання й обробки візуальних зображень. Java містить усі функції високого рівня, необхідні в

індустрії робототехніки, особливо коли йдеться про штучний інтелект, машинне навчання, пошук та нейронні алгоритми. Мова Java «приховує» основні функції пам'яті від програміста, що полегшує програмування. Теорія використання цієї мови полягає в тому, що можна використовувати один і той самий код на різних машинах, завдяки віртуальній машині Java, хоча це дещо уповільнює виконання коду.

LISP і Prolog – мови логічного програмування, які використовують для програмування штучного інтелекту (ШІ), що останнім часом стрімко набирає популярності. Багато важливих розділів ROS написані засобами LISP, а Prolog використовувався для програмування Watson AI від IBM. Хоча нині програмувати ШІ можна і засобами інших мов програмування, однак LISP і Prolog залишаються в основі деяких реалізацій ШІ і, безумовно, заслуговують на своє місце в цьому списку.

C#/.NET – мова програмування від Microsoft, яка використовується для розробки у Visual Studio. Для робототехніки C#/.NET зазвичай використовується в програмуванні на рівні портів і сокетів. C# може використовуватися для впровадження нейронних мереж. Приклади включають Netduino, FEZ Rhino тощо [3].

Отже, програмування є одним з основних навиків, який необхідно розвивати для роботи в сфері робототехніки. Мови програмування мають глибокий вплив на інтеграцію роботів у промислові машини, а роботизоване програмне забезпечення відіграє ключову роль у виконанні складних операцій і точної функціональності.

Список використаних джерел

1. Different Types of Robot Programming Languages. URL: <https://www.plantautomation-technology.com/articles/different-types-of-robot-programming-languages>.
2. Owen-Hill A. What is the Best Programming Language for Robotics? URL: <https://blog.robotiq.com/what-is-the-best-programming-language-for-robotics>

3. Davidson A. Must learn top programming languages for robotics in 2022.
URL: <https://codersera.com/blog/top-programming-languages-for-robotics/>

Ключові слова: робототехніка, мови програмування, програмування робота.

Ключевые слова: робототехника, языки программирования, программирование робота.

Keywords: robotics, programming languages, robot programming.

Науковий керівник: к.т.н., доц. Трофименко О. Г.

Титиевский Виталий Олегович

*Национальный университет «Одесская юридическая академия», студент
2-го курса факультета кибербезопасности и информационных технологий*

БЫСТРОДЕЙСТВИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

Ни для кого не секрет, что существует огромное количество языков программирования. С каждым годом их становится всё больше и больше. Логичным будет вопрос: «Какой язык программирования лучше?» Естественно, есть много критериев, по которым можно выбрать тот или иной язык, при этом стоит учитывать, что каждый язык хорош по-своему и у каждого программиста есть свои предпочтения. Одним из объективных критериев является критерий быстродействия языка.

Скорость языка программирования зависит от нескольких факторов: кроссплатформенность, компилируемость, статическая и динамическая типизация [1]. Проанализируем их.

Первый фактор – наличие компилятора, то есть возможности преобразовать программный код в набор инструкций для данного типа процессора (машинный код) и далее в исполняемый файл, который может быть запущен на выполнение