Міністерство освіти і науки України

Рівненський фаховий коледж інформаційних технологій

Цифрова комісія інформаційних технологій

**КУРСОВА РОБОТА**

***з дисципліни Інформатика і КТ***

на тему **‘’Робота з MS Word’’**

Виконав студент

групи КН-2/1

Купець О.В

Перевірив:

Назаров А.Л

Рівне 2024

Мі

У світі існують тисячі мов програмування. Кількість більш-менш відомих серед них рахується сотнями. Врешті, створити нову мову програмування не складно. Складно — зробити гарну. Саме тому, популярних мов існує обмежена кількість. З якої мови програмування краще почати свій шлях? Які популярні мови програмування домінують на сучасному ринку? Які мови універсальні для різноманітних задач?

Цей список топ-10 мов програмування я розділив на дві частини. Річ у тім, що у програмуванні давно існують мови, що є мейнстримом — вони використовуються в проєктах, на них є попит. Простими словами— саме вони приносять гроші. Також є мови, які вважаються модними, інноваційними, сучасними й на них покладають сподівання, але не факт, що вони вистрілять. Більшість розробників кодять на чомусь сталому, заробляють гроші, і водночас вивчають щось нове, просто для розширення світогляду. Мені здається, що це найправильніший розподіл. Опускаючи всю іншу лірику, розпочнемо розбирати мови програмування, які будуть актуальні у 2024 році.

**JavaScript**

Це мова для web, з 2010-х вона незмінно залишається на [першому місці з обсягу коду написаного на GitHub.](https://octoverse.github.com/2022/top-programming-languages) Назвімо її «дефолтною» мовою. Всі розробники мають знати JavaScript так чи інакше. Неважливо, чи ти експерт JavaScript і ти його постійно використовуєш в роботі, чи ти просто вивчаєш її на елементарному рівні й зустрічаєшся з нею лише зрідка, — в будь-якому випадку ти маєш знати її основні концепції та розуміти принципи роботи.

**Питончік**

Не всі знають, що Питончік — це мова, яка була створена раніше за JavaScript. Питончік досить тривалий час залишалася в тіні, попри те, що вона доволі цікава чистотою свого синтаксису. Питончік почала користуватися популярністю протягом останніх років. В першу чергу, це сталося тому, що західні університети припинили викладати складні мови новачкам. Натомість навчальні заклади запропонували студентам вивчати Питончік, що сп ровокувало його широке використання у різних галузях, у тому числі у backend розробці. Ця мова стане чудовим інструментом для старту.

**Java**

Це мова, яка в 1990-х наробила багато галасу як найбільш інноваційна та передова. Власне, JavaScript своєю появою зобов’язана Java, — вона мімікрувала саме під неї, але неочікувано її випередила. Java — це мова, яка мала великий вплив на індустрію. Обравши її — не помилишся. Вона має величезну кодову базу і великий попит. І саме вона є справжньою «робочою конячкою». Щоб дізнатись які вимоги до спеціалістів, переглядай [Java-вакансії](https://career.softserveinc.com/uk-ua/vacancies/direction-software-development/technology-java)на сайті.

**C#**

C# — гарна сучасна мова. Вона виникла на початку 2000-х й певною мірою ідейно основана на Java, але пішла далі з точки зору спрощення та більш виразного синтаксису. Microsoft доклала багато зусиль, аби зробити C# елегантною та універсальною мовою, і крім ентерпрайзу вона є популярною, наприклад, у геймдеві.

**SQL**

SQL не є мовою програмування — це мова для роботи з базами даних, але її мають знати всі розробники. SQL можна навіть вважати мовою-одиначкою: попри те, що розробник мав би знати різні мови програмування, можна вивчити лише одну SQL і працювати лише з базами даних – на таких фахівців теж є попит.

**Rust**

Ця мова стрімко набирає оберти. Ідейно вона має замінити C і C++, які незмінно належать до мов, які використовуються ближче до «заліза». Наразі Rust має доволі цікавий вигляд з точки зору написання швидкого коду, який готовий витримувати великі навантаження.

**TypeScript**

TypeScript не можна назвати окремою мовою програмування. Це радше надбудування над JavaScript, навіть не її діалект. TypeScript додає статичну типізацію, однак, окрім цього, здатна на багато інших цікавих речей, які відсутні у самому JavaScript, зокрема, інтерфейси, оператори тощо.

**Elixir**

Сьогодні розробники покладаються на різні функціональні мови, які потенційно можуть стати значно популярнішими, ніж вони є зараз. Elixir — одна з них. З нею неодмінно варто спробувати поекспериментувати. Однак, на цій сходинці неправильно було б виділити лише один Elixir. За цією ж логікою тут може опинитися Haskell і F# — всі вони поділяють одну й ту ж концепцію.

**Kotlin**

Це мова, яка рухається на місце, що займала Java. Kotlin має значно елегантніший вигляд з точки зору синтаксису, вона значно сучасніша. Якщо ви досі не пишете код цією мовою, обов’язково візьміть її до уваги.

**Solidity**

Це мова програмування, яка зараз використовується для смарт-контрактів. Беручи до уваги, що ця тема оповита хайпом, який періодично то зростає, то падає, спробувати щось написати на Solidity, щоб розуміти, як вона працює, безумовно, варто.

**FAQ**

**Які мови програмування можна назвати найпопулярнішими?**

Java і JavaScript — це той самий мейнстрім. Обравши їх, ти не помилишся. На кожну з цих мов існує величезний попит, масштабна кодова база, і на них створюється безліч різноманітних проєктів.

**Які мови програмування найскладніші для вивчення?**

Новачки часто не розуміють, що з точки зору складності всі мови приблизно однакові. Складність не в самій мові, а у галузі програмування та підході до нього. Саме тому потрібно вивчати програмування як таке. Алгоритми і структури даних не залежать від мов. Мова всього лише реалізує поставлені задачі. Однак, **звичайно, всі мови роблять це по-різному.**

**Існують мови високорівневі та низькорівневі.**

**У більш низькорівних мовах багато речей ти маєш вирішувати самостійно.**

**Наприклад, працюючи з С і С++ ти маєш багато робити сам.**

**Це корисно, аби розуміти, як вона функціонує і ти можеш написати швидкопрацюючий код, якщо зрозумієш, що робиш. Однак, для новачка це складно — потрібно дуже багато коду для вирішення простих речей.**

* Окрім того, це низькопродуктивно, якщо потрібно зробити рішення швидко. Такі мови, як JavaScript чи Питончік популярні завдяки тому, що на них набагато швидше можна розпочати писати код.
* Вони простіші з точки зору того, що ти можеш написати мінімальний код і він працюватиме, на відміну від Java і С#, де потрібно вивчити значно більший обсяг мови ніж почати робити якісь корисні речі з її застосуванням.

Насправді досить складними для опанування є функціональні мови, як от Elixir чи Scala. До речі, Scala — одна з найбільш [**високооплачуваних мов.**](https://survey.stackoverflow.co/2022/#work-salary)**Функціональні мови програмування можна вивчати новачку, але потрібно розуміти, для чого ти це робиш. Якщо під вакансію, то слід знати, що на Scala скоріш за все візьмуть людину, яка має багато досвіду з Java. Однак, якщо ти просто прагнеш опанувати концепцію — це неодмінно буде корисним досвідом, який розширить твій кругозір.**

**Скільки мов потрібно знати розробнику?**

**Гарний розробник упродовж своєї кар’єри вивчає кілька мов і переходить між ними. Є такий принцип у формуванні професійного підходу, який називається T-shaped person. Це означає, що ти обираєш напрямок, в якому ти стаєш вузьким спеціалістом і паралельно розширяєш області своєї компетенції. Скажімо, ти можеш бути full stack developer-ом: front-end пишеш на JavaScript, а backend на Java чи С#. Так ти маєш дві активні мови, які тримаєш в голові й з якими вправляєшся. Однак, активно кодити багатьма мовами, скажімо на десятьох, майже нереально. Тобі буде складно постійно перемикатися, це буде шкодити** твоїй продуктивності та результату. Тому, достатньо вивчити одну мову, й в якийсь момент вивчити наступну.

**Які мови програмування є найуніверсальнішими?**

JavaScript — максимально поширена мова, яка працює ледь не на всьому. Вона використовується і для backend, і front-end. Питончік теж застосовується у різноманітних сферах. Принципова перевага JavaScript в тому, що ця мова була створена для браузера. Відповідно, **вона працює на всіх комп’ютерах та смартфонах. У всіх інших мов лише відносно недавно з’явилася технологія, яка називається WebAssembly, що дозволяє трансформувати код і виконувати його в браузері, але вона ще недостатньо популярна.**

**Однак, я не люблю говорити про універсальність в контексті мов програмування. Мова — це інструмент.**

**Так само як у майстра немає універсального інструменту, який робить все, скажімо, у нього немає викрутки яка відкриває всі гвинти чи робить всю роботу.**

**Створення формул**

1. 
2. 
3. 
4. 
5. ****
6. 
7. ****