

Особливості вимог до спеціаліста з високопродуктивного комп'ютерингу

Вимоги	High Performance Engineer
Основні навички	Глибоке розуміння архітектури комп'ютера, оптимізація коду для високої продуктивності, розуміння принципів роботи комп'ютерних систем та їх взаємодії.
Мови програмування	Має глибокі знання мов програмування, які зазвичай включають C++, Java, асемблер, може використовувати мови з низькорівневого рівня, такі як CUDA або OpenCL.
Розробка ПЗ	Спеціалізується на оптимізації виконання програмного забезпечення, розробці бібліотек та інструментів для високопродуктивних застосунків.
Види проектів	Розробка високопродуктивних додатків, таких як системи обробки даних в реальному часі, алгоритми машинного навчання, ігри тощо.
Знання архітектури	Глибоке розуміння архітектурних принципів комп'ютера, включаючи роботу з пам'яттю, кеш-пам'яттю, оптимізацію для конкретних архітектур процесорів тощо.
Оптимізація продуктивності	Спеціалізується на оптимізації продуктивності на низькорівневому рівні, включаючи оптимізацію пам'яті, виконання операцій з даними, оптимізацію вводу/виводу тощо.
Пропоновані позиції на ринку	High Performance Computing Engineer Systems Performance Engineer Performance Optimization Engineer Parallel Computing Engineer Systems Software Engineer GPU Engineer Computational Scientist Firmware Engineer (якщо спеціалізується на оптимізації для вбудованих систем) Compiler Engineer Data Engineer (якщо спеціалізується на оптимізації обробки даних)

Аналіз вимог ринку праці високопродуктивного обчислення

Протягом останніх років спостерігається загальна тенденція до зростання інтеграції різних технологічних підходів, а також до більшої фокусування на розробці ефективних та масштабованих рішень. Крім того, спеціалізації стають більш гнучкими, що дозволяє розробнику високопродуктивних рішень швидше адаптуватися до нових вимог та технологій ринку. Загалом, основною відмінністю між розробниками програмного забезпечення та інженерами високопродуктивних обчислень є використання різних підходів. Розробники програмного забезпечення залишаються ключовими гравцями на ринку праці, і здатні адаптуватися до нових технологій та спеціалізацій які є важливим елементом успіху в цій динамічній галузі, тоді коли інженери високопродуктивних обчислень мають конкретну спеціалізацію в певній області і здатні вдосконалювати, розвивати та здійснювати постійний ріст розроблених додатків та проектів.

Вакансії, що набуватимуть популярності згідно схеми [Gartner](#) для інженерів високопродуктивних обчислень:

Emergent AI

- **(AI simulation) Моделювання штучного інтелекту** – це комбіноване застосування технологій штучного інтелекту та моделювання для спільної розробки агентів штучного інтелекту та змодельованих середовищ, у яких їх можна навчати, тестувати та іноді розгортати.

- **(Casual AI) Причинно-наслідковий штучний інтелект** визначає та використовує причинно-наслідкові зв'язки, щоб вийти за рамки прогнозних моделей на основі кореляції та перейти до систем штучного інтелекту, які можуть призначати дії більш ефективно та діяти більш автономно.
- **(Federated machine learning) Об'єднане машинне навчання** спрямоване на навчання алгоритму машинного навчання без явного обміну зразками даних, що забезпечує кращу конфіденційність і безпеку.
- **(Graph data science) Наука про графічні дані (GDS)** — це дисципліна, у якій методи науки про дані застосовуються для побудови структур даних для визначення поведінкових характеристик, які можна використовувати для побудови прогнозних і приписних моделей.
- **(Neuro-symbolic AI) Нейросимволічний штучний інтелект** — це форма складеного штучного інтелекту, яка поєднує методи машинного навчання (ML) і символічні системи для створення більш надійних і надійних моделей штучного інтелекту.
- **(Reinforcement learning RL) Навчання з підкріпленням (RL)** – це тип ML, де навчальна система отримує навчання лише з точки зору позитивного зворотного зв'язку (винагороди) і негативного зворотного зв'язку (покарання).

Досвід розробника (DevX) (Developer experience (DevX))

- **(AI-augmented software engineering) Розробка програмного забезпечення**, доповнена штучним інтелектом, використання технологій ШІ та обробки природної мови (NLP), щоб допомогти розробникам програмного забезпечення створювати, доставляти та підтримувати програми.
- **(API-centric Saas) SaaS**, орієнтований на API, хмарна служба додатків, розроблена з використанням програмних запитів/відповідей або інтерфейсів на основі подій (API) як основних методів доступу.
- **GitOps, тип замкнутої системи керування для хмарних програм.**
- **(Internal developer portals) Внутрішні портали для розробників**, які дають змогу самостійно знаходити та отримувати доступ до ресурсів у складних середовищах розробки програмного забезпечення в хмарі.
- **(Open-source program office) Офіс програм з відкритим кодом (OSPO)**, центр компетенції для побудови стратегій управління, управління, просування та ефективного використання програмного забезпечення з відкритим кодом (OSS) і даних або моделей з відкритим кодом.

Всепроникна хмара (Pervasive cloud)

- **(Automated FinOps) Розширений FinOps**, який застосовує традиційні концепції DevOps гнучкості, безперервної інтеграції та розгортання, а також відгуків кінцевих користувачів для фінансового управління, бюджетування та зусиль з оптимізації витрат.
- **(Cloud development environments CDEs) Хмарні середовища розробки (CDE)**, що забезпечують віддалений готовий до використання доступ до хмарного середовища розробки з мінімальними зусиллями для налаштування та налаштування.
- **(Cloud sustainability) Хмарна стабільність**, використання хмарних сервісів для досягнення переваг у стійкості в економічних, екологічних і соціальних системах.
- **(Cloud-native) Власне в хмарі**, що відноситься до чогось, створеного для оптимального використання або реалізації характеристик хмари, які є частиною початкового визначення хмарних обчислень, і включає можливості, що надаються як послуга.
- **Cloud-out to edge**, архітектурна конструкція, у якій централізовано кероване хмарне середовище, як правило, гіпермасштабована хмара, надає можливості хмарного сервісу, які поширюються на периферійні середовища.
- **WebAssembly (Wasm)**, легка машина віртуального стеку та формат двійкового коду, призначена для підтримки безпечних високопродуктивних програм на веб-сторінках.
- **Екзаскейл-системи:** Розробка та впровадження екзаскейл-систем, які здатні виконувати екзафлопс обчислень в секунду. Це дозволить вирішувати надзвичайно складні задачі в області науки, інженерії та інших галузей.
- **Гетерогенні обчислювальні системи:** Збільшення використання графічних процесорів (GPU) та інших прискорювачів для підвищення ефективності паралельних обчислень.

- **Розвиток великих обчислювальних хмар:** Забезпечення великих потужностей обчислень у формі обчислювальних хмар, які дозволяють використовувати ресурси в режимі реального часу за необхідності.
- **Архітектурна оптимізація для енергоефективності:** Зусилля в напрямку створення енергоефективних обчислювальних систем, оскільки це є ключовим аспектом при масштабуванні обчислень.
- **Розширення використання квантових обчислювачів:** Дослідження та розробка алгоритмів, які можуть бути виконані на квантових обчислювачах, що відкриває нові можливості для вирішення певних класів завдань.
- **Використання штучного інтелекту в управлінні ресурсами:** Впровадження методів машинного навчання та штучного інтелекту для ефективного управління ресурсами високопродуктивних систем, що дозволяє підвищити продуктивність та ефективність використання ресурсів.
- **Розвиток кіберфізичних систем:** Інтеграція високопродуктивних обчислень у кіберфізичні системи, що об'єднують фізичний світ з обчислювальними процесами.
- **Розширення застосувань у важких галузях:** Застосування високопродуктивних систем у важких галузях, таких як кліматичні дослідження, медичні обчислення, обробка великих обсягів даних тощо.

Розробник високопродуктивних обчислень у взаємодії з різними галузями та використання різних технологічних стеків може бути спрямована на досягнення різних цілей, таких як покращення функціональності, вдосконалення користувацького досвіду, оптимізація процесів та вирішення конкретних завдань. Ось кілька аспектів взаємодії інженера високопродуктивних обчислень з іншими галузями та використання різних технологічних стеків:

Міжгалузева комунікація:

- **Бізнес-аналітика та менеджмент:** Розробники повинні зрозуміти бізнес-вимоги та потреби, спілкуватися з бізнес-аналітиками та менеджерами для визначення та розробки програмного продукту, який відповідає стратегічним цілям компанії.
- **Дизайнери та UX/UI фахівці:** Забезпечення зручного та естетичного інтерфейсу для користувачів, співпраця щодо дизайну, взаємодія з елементами інтерфейсу.
- **QA і тестувальники:** Розробники взаємодіють з фахівцями з тестування для впровадження та удосконалення тестових стратегій та автоматизації тестування.

Використання різних технологічних стеків:

- **Front-end і Back-end розробники:** Комунікація та співпраця між розробниками, які відповідають за клієнтську та серверну частини додатка.
- **DevOps-інженери:** Розробники співпрацюють з DevOps-інженерами для реалізації та автоматизації процесів поставки, впровадження та моніторингу додатків.
- **Спеціалісти з баз даних:** Взаємодія щодо проектування та оптимізації баз даних, вибір та впровадження відповідних технологій для зберігання та обробки даних.

Розвиток специфічних рішень:

- **Науковці та дослідники:** Взаємодія з фахівцями в різних галузях науки для розробки та впровадження новаторських алгоритмів, методів та технологій.
- **Спеціалісти з інженерії та виробництва:** Співпраця щодо розробки програмного забезпечення для автоматизації та оптимізації виробничих процесів.
- **Отже,** взаємодія з різними галузями та використання різних технологічних стеків допомагає розробникам створювати комплексні та ефективні рішення, які відповідають вимогам різних індустрій і користувачів.