

Задачи для Летней интернатуры 2019 года

Задача 1.	Разработка документации для установочного фреймворка программного обеспечения Intel®. ..2
Задача 2.	Разработка приложения, осуществляющего видео стрим на игровой сервис Twitch с помощью девайсов на архитектуре Intel®.....2
Задача 3.	Разработка структур данных для энергонезависимой памяти2
Задача 4.	Разработка рекомендаций по оптимизации приложений для современных аппаратных решений Intel с использованием Intel® Advisor.....3
Задача 5.	Исследование и разработка распределенного Градиентного Бустинга в Intel® DAAL.....3
Задача 6.	Расширение функциональности, поддерживаемой Градиентным Бустингом в Intel® DAAL3
Задача 7.	Исследование и разработка методов предсказания вероятностей классов в задачах классификации.....4
Задача 8.	Адаптация программного интерфейса для автоматизации анализа приложений (Python API) с использованием Intel® Advisor для языка Python 34
Задача 9.	Разработка трансформаций в OpenVINO™ Model Optimizer4
Задача 10.	Использование компьютерного зрения в системах реального времени для промышленной автоматизации4
Задача 11.	Использование камер Intel® RealSense™ в системах реального времени для промышленной автоматизации5
Задача 12.	Разработка алгоритма с использованием глубокого обучения для задачи по оценке позы человека 5
Задача 13.	Поиск эффективных методов сжатия свёрточных нейронных сетей.....5
Задача 14.	Разработка алгоритма с использованием глубокого обучения для решения практических задач. 5
Задача 15.	Разработка и поддержка платформы для тренировки алгоритмов глубокого обучения на кластере 6
Задача 16.	Разработка фреймворка для построения, анализа и отладки набора данных с разметкой для алгоритмов глубокого обучения.....6
Задача 17.	Графовые оптимизации для глубокого обучения на платформе Movidius.....6
Задача 18.	Оптимизация слоя сети глубокого обучения для платформы VPU ускорителей6
Задача 19.	Реализация OpenCV camera multiplexing API для Linux*.....6
Задача 20.	Улучшение метрик качества VPU плагинов Inference Engine7
Задача 21.	Улучшение алгоритмов детектирования QR-кодов.7
Задача 22.	Подготовка обучающих примеров по компьютерному зрению для freeCodeCamp.org с помощью OpenCV.js 7
Задача 23.	Портирование алгоритмов компьютерного зрения на OpenCV Graph API7
Задача 24.	Система интеграции для открытого зоопарка моделей.....8
Задача 25.	Внедрение алгоритмов компьютерного зрения в приложение8
Задача 26.	Портирование демо из продукта OpenVINO™ toolkit с языка C++ на Python. Создание новых демо 8
Задача 27.	Исследование и реализация поддержки новых топологий нейронных сетей в OpenVINO™8
Задача 28.	Оптимизация алгоритма машинного обучения под GPU8

Задача 29.	Разработка инструмента для исследования HPC приложений с использованием Intel® Advisor ...	9
Задача 30.	Автоматизация методологии характеристики MPI-приложения на основе инструментов Intel....	9
Задача 31.	Оптимизация параллельных алгоритмов стандартной библиотеки C++ для гетерогенных архитектур	9
Задача 32.	Сравнительный анализ компиляторов на процессорах Intel	9
Задача 33.	Миграция и рефакторинг систем автоматического построения сэмплов и запуска тестов OpenVX	9
Задача 34.	Профилировка Spark-приложений на Hadoop-кластере с помощью Intel® VTune Amplifier	10
Задача 35.	Анализ производительности параллельных алгоритмов стандартной библиотеки C++ (Parallel STL)	10
Задача 36.	Разработка и оптимизация инструментов и алгоритмов в OpenVINO™	10
Задача 37.	Создание основанного на веб-технологиях помощника для визуализации и моделирования соответствия ядер CPU и процессов MPI-приложения	10
Задача 38.	Разработка эффективных алгоритмов для обработки естественных языков (NLP).....	10

Задача 1. Разработка документации для установочного фреймворка программного обеспечения Intel®

Описание Разрабатываемое программное обеспечение должно надежно устанавливаться на различные платформы (Windows*, Linux*, macOS*, FreeBSD*). Для этого в Intel разработан фреймворк, который позволяет упростить и автоматизировать создание установочных пакетов. Необходимо разработать руководство пользователя, которое поможет разработчикам воспользоваться данным фреймворком.

Документ должен описывать:

1. Подготовку продуктового контента, удовлетворяющего требованиям проверки целостности (спецификации BOM)
2. Создание установочных модулей для различных платформ (MSI, RPM, DEB, DPKG и т.д.)
3. Создание репозитория для различных платформ (YUM/APT и т.д.)
4. Настройку фреймворка для создания и проверки установочного пакета.

Требования к претенденту

- Отличное знание английского языка
- Знание основ программирования и языков гипертекста (HTML, CSS)

Приветствуются знания программных продуктов Intel и опыт работы с различными операционными системами.

Задача 2. Разработка приложения, осуществляющего видео стрим на игровой сервис Twitch с помощью девайсов на архитектуре Intel

Описание Необходимо разработать приложение, поддерживающее современную архитектуру Intel, позволяющее стримить современные игры на популярный сервис Twitch. Основная функция приложения - возможность стримить популярные компьютерные игры на Twitch. Для разработки приложения будут использоваться передовые технологии и программные продукты компании Intel.

Требования к претенденту C/C++
Знание современных видео кодеков будет плюсом.

Задача 3. Разработка структур данных для энергонезависимой памяти

Описание Персистентная память - это тип энергонезависимой памяти с побайтовой адресацией, как и DRAM, и возможностью сохранять своё состояние во время перезапуска процесса или перезагрузки системы. PMDK - это набор библиотек, предоставляющих разработчикам приложений доступ к персистентной памяти. Новый тип памяти позволяет создавать структуры

данных с временем жизни, превышающим время жизни процесса. Но существующие структуры данных, разработанные для энергозависимой оперативной памяти, не подходят для новой памяти и должны быть портированы. Структуры данных для энергонезависимой памяти должны поддерживать согласованность данных в случае системных ошибок и сбоев.

Требуется расширить PMDK новыми структурами данных для энергонезависимой памяти.

- Требования к претенденту**
- C++
 - Структуры данных и алгоритмы
 - Метапрограммирование
 - ООП

Задача 4. Разработка рекомендаций по оптимизации приложений для современных аппаратных решений Intel с использованием Intel® Advisor

Описание Intel® Advisor – инструмент для создания и оптимизации высокопроизводительных приложений для современных вычислительных архитектур, входящий в состав Intel® Parallel Studio. В рамках задачи необходимо исследовать проблему, связанную с производительностью векторизованного кода на современном аппаратном оборудовании Intel, разработать алгоритмы её детектирования и интегрировать их в Intel® Advisor.

- Требования к претенденту**
- Понимание различных техник оптимизации кода.
Желательно хорошее знание C++.
Знание x86 архитектуры и ассемблера будет дополнительным преимуществом.

Задача 5. Исследование и разработка распределенного Градиентного Бустинга в Intel® DAAL

Описание Одним из самых популярных алгоритмов машинного обучения, который часто применяется для обучения на «больших данных», является Градиентный Бустинг. Библиотека Intel® DAAL (предоставляет решения для машинного обучения, оптимизированные для Intel® архитектур) реализует примитивные блоки для организации вычислений на распределенной памяти для этого алгоритма. При этом для коммуникации между узлами кластера может быть использована любая библиотека для распределенных вычислений - Intel® MPI, Apache Spark*, Dask.

Требуется исследовать возможные распределенные топологии для Градиентного Бустинга с целью оптимизации времени вычислений, написать примеры использования распределенной версии алгоритма с использованием перечисленных библиотек.

- Требования к претенденту**
- C++ и Python
 - Опыт в написании распределенных приложений
 - Базовые знания машинного обучения
 - XGBoost/LightGBM/CatBoost будет дополнительным плюсом

Задача 6. Расширение функциональности, поддерживаемой Градиентным Бустингом в Intel® DAAL

Описание Одним из самых популярных алгоритмов классического машинного обучения является Градиентный Бустинг деревьев, особенно он популярен среди участников соревнований по анализу данных, таких как Kaggle, из-за своих хороших результатов.

Библиотека Intel® DAAL (предоставляет решения для машинного обучения, оптимизированные для Intel® архитектур) имеет хорошо оптимизированную реализацию данного алгоритма, но поддерживает только подмножество параметров, поддерживаемых другими библиотеками, такими как XGBoost, LightGBM и CatBoost.

Требуется исследовать параметры, которые на данный момент не поддерживаются в Intel® DAAL, их влияние на точность предсказания и частоты их использования в анализе данных. Необходима программная реализация наиболее полезных параметров, отсутствующих в Intel® DAAL и интеграция в нее, а также сравнение полученных результатов с другими библиотеками.

- Требования к претенденту**
- C/C++
 - Хороший математический аппарат
 - Базовые знания машинного обучения
- Знакомство с одной из библиотек XGBoost/LightGBM/CatBoost будет дополнительным плюсом

Задача 7. Исследование и разработка методов предсказания вероятностей классов в задачах классификации

- Описание** Многие алгоритмы машинного обучения, решающие задачу классификации, могут предсказывать не только номер класса для конкретного наблюдения, но и уверенность обученного классификатора в данном ответе т.е. вероятности каждого исхода (например, логистическая регрессия, решающие деревья).
Требуется исследовать методы подсчета этих вероятностей в задаче классификации. Необходима программная реализация выбранных методов для нескольких алгоритмов, их интеграция в Intel® DAAL (предоставляет решения для машинного обучения, оптимизированные для Intel® архитектур) и сравнение результатов классификации и производительности с другими библиотеками машинного обучения.
- Требования к претенденту**
- C/C++
 - Хороший математический аппарат
 - Базовые знания машинного обучения
- Знакомство с фреймворками для машинного обучения и Python будет дополнительным плюсом

Задача 8. Адаптация программного интерфейса для автоматизации анализа приложений (Python API) с использованием Intel® Advisor для языка Python 3

- Описание** Intel® Advisor – инструмент для создания и оптимизации высокопроизводительных приложений для современных вычислительных архитектур, входящий в состав Intel® Parallel Studio. Необходимо адаптировать программный интерфейс для загрузки и детального анализа полученных профилей для языка Python 3, наравне с имеющейся поддержкой для Python 2.
- Требования к претенденту**
- Хорошее знание C++,
 - Знание Python.
- Дополнительным преимуществом будет знание boost::python, опыт работы со встроенным Python.

Задача 9. Разработка трансформаций в OpenVINO™ Model Optimizer

- Описание** Deep Learning Deployment Toolkit (DLDT) -- это часть хорошо известного и быстро развивающегося набора компонент OpenVINO™. Model Optimizer -- это одна из ключевых частей DLDT, которая создана для конвертирования и оптимизации моделей глубокого обучения из разных фреймворков, таких как Caffe*, TensorFlow* и MXNet для их быстрого исполнения. Постоянно растущее количество различных моделей, которые требуется поддержать в DLDT, приводит к необходимости реализации всё большего числа алгоритмов трансформаций моделей.
Требуется изучить внутренние детали устройства Model Optimizer и Inference Engine, реализовать и добавить в DLDT несколько новых трансформаций:
- (1) Поддерживать новые модели глубокого обучения,
 - (2) Поддерживать новые операции из фреймворков для глубокого обучения (например, новые операции из TensorFlow* или ONNX).
 - (3) Оптимизировать уже поддерживаемые модели и комбинации операций в платформенно-независимой форме.
- Требования к претенденту**
- Хорошее знание C++ и Python, опыт работы в реальных проектах
 - Опыт работы с Git
 - Базовые знания о свёрточных сетях
 - Хорошая математическая подготовка (теория графов, линейная алгебра)
- Дополнительным преимуществом является:
- Опыт работы с Pytorch, TensorFlow* или каким-либо другим фреймворком глубокого обучения
 - Знания компиляторных технологий

Задача 10. Использование компьютерного зрения в системах реального времени для промышленной автоматизации

- Описание** OpenVINO™ Toolkit (<https://software.intel.com/en-us/openvino-toolkit>) позволяет ускорить процесс создания систем компьютерного зрения на различных аппаратных платформах Intel. В

настоящее время toolkit не достаточно широко применяется в системах жесткого реального времени для промышленной автоматизации. Необходимо провести исследование возможностей использования OpenVINO™ Toolkit в задачах реального времени, разработать прототип, демонстрирующий один из способов использования и интегрировать его с библиотекой улучшающей время отклика системы.

Требования к претенденту Навыки программирования на C/C++ и желательно Python в ОС Linux*
Желателен опыт в задачах компьютерного зрения с использованием OpenVINO™ toolkit или OpenCV

Задача 11. Использование камер Intel® RealSense™ в системах реального времени для промышленной автоматизации

Описание 3D-камеры и библиотеки Intel® RealSense™ позволяют разрабатывать приложения компьютерного зрения, использующие датчик глубины. В данной задаче нужно разработать демо приложение, которое позволит контролировать окружающую обстановку и управлять действиями промышленного робота (например, отключать устройство при нахождении человека слишком близко). При разработке приложения нужно будет исследовать как совместить два цикла на одной платформе: компьютерное распознавание и промышленное управление в режиме жесткого реального времени

Требования к претенденту Навыки программирования на C/C++ и желательно Python в ОС Linux*.
Желателен опыт в задачах компьютерного зрения с использованием 3D-камер, Intel® RealSense™ SDK или OpenCV.

Задача 12. Разработка алгоритма с использованием глубокого обучения для задачи по оценке позы человека

Описание Задача определения позы человека состоит в нахождении координат её ключевых точек: плечи, локти, запястья, шея и т.д., пример <https://www.youtube.com/watch?v=pW6nZxeWIGM>. Необходимо разработать алгоритм для определения позы человека в трехмерном пространстве с использованием глубокого обучения.

Требования к претенденту

- Python
- Знания в области машинного и глубокого обучения
- Английский на уровне чтения научных статей

Будет плюсом

- Опыт работы с PyTorch
- Стажировка в [ITLab](#) или похожих лабораториях
- Участие в соревнованиях по машинному обучению: kaggle, topcoder, drivendata, yandex algorithm, и т.д.
- Чтение статей из <http://www.arxiv-sanity.com/>

Задача 13. Поиск эффективных методов сжатия свёрточных нейронных сетей

Описание Требуется применять и экспериментировать с последними разработками в области оптимизации сетей в применении к современной архитектуре, чтобы достичь высокой эффективности и учесть особенности аппаратной платформы Intel.

Требования к претенденту

- Python
- Английский на уровне чтения статей

Будет плюсом:

- Знания в области машинного и глубокого обучения
- Участие в соревнованиях по машинному обучению: kaggle, topcoder, drivendata, yandex algorithm, и т.д.
- Научная деятельность в университете

Задача 14. Разработка алгоритма с использованием глубокого обучения для решения практических задач.

Описание Разработка алгоритма с использованием глубокого обучения для решения практических задач.

- Требования к претенденту**
- Python
 - Английский на уровне чтения статей
- Будет плюсом:
- Знания в области машинного и глубокого обучения
 - Участие в соревнованиях по машинному обучению: kaggle*, topcoder*, drivendata*, yandex* algorithm, и т.д.
 - Научная деятельность в университете.

Задача 15. Разработка и поддержка платформы для тренировки алгоритмов глубокого обучения на кластере

- Описание** Наличие огромного количества вычислительных ресурсов без удобного способа их использовать параллельно многими пользователями существенно тормозит исследования в области глубокого обучения. Необходимо разработать и поддерживать платформу для управления ресурсами кластера, которая базируется на платформе с открытым исходным кодом Microsoft Platform for AI <https://github.com/Microsoft/pai>.
- Требования к претенденту**
- Опыт программирования на Python 3.x и Node.js
 - Чтение технической документации на английском языке
- Будет плюсом - знания в области машинного и глубокого обучения

Задача 16. Разработка фреймворка для построения, анализа и отладки набора данных с разметкой для алгоритмов глубокого обучения

- Описание** Данные - это стратегический ресурс для решения многих проблем. Так, для алгоритмов глубокого обучения, которые активно используются и разрабатываются внутри Intel, данные - это самостоятельный продукт, без которого невозможно достичь должного уровня качества и удовлетворить требования заказчиков. Для создания наборов данных необходимо наличие удобных инструментов. В данной задаче необходимо разработать и внедрить в существующую инфраструктуру инструменты для построения, анализа и отладки данных для алгоритмов глубокого обучения. Хорошие примеры инструментов, которые уже существуют в этой области: Facets* и What-If tool*.
- Требования к претенденту**
- Опыт программирования на Python 3.x
 - Чтение технической документации на английском языке
- Будет плюсом - знания в области машинного и глубокого обучения

Задача 17. Графовые оптимизации для глубокого обучения на платформе Intel® Movidius™

- Описание** Оптимизация компилятора графов. Компилятор графов преобразует высокоуровневое описание нейронной сети в низкоуровневое представление для оптимального вычисления на целевом процессоре. В нашем случае, на процессоре Intel Movidius, специализированном для нейронных сетей.
- Требования к претенденту**
- C++
- Понимание нейронных сетей будет плюсом

Задача 18. Оптимизация слоя сети глубокого обучения для платформы VPU ускорителей

- Описание** Необходимо написать эффективную реализацию определенного слоя сети, улучшив таким образом время вычисления результата нейронной сети на платформе VPU ускорителей.
- Требования к претенденту**
- Требуется:
- C/C++
 - Английский язык на уровне чтения документации
- Будет плюсом:
- Знания в области глубокого обучения
 - Знания в области оптимизации алгоритмов

Задача 19. Реализация OpenCV camera multiplexing API для Linux*

- Описание** В данный момент в OpenCV нет эффективного API для работы с несколькими камерами одновременно. Пользователи вынуждены использовать непертируемые системные API или

мириться с неоптимальной производительностью. Было предложено дополнение к существующему OpenCV API (<https://github.com/opencv/opencv/issues/12077>), а также была сделана независимая реализация с использованием Linux* v4l2 API. Требуется реализовать данное OpenCV API для платформы Linux*. Примером приложения, которому необходимо данное API, является Multichannel Face Detection demo и его текущие сценарии использования камер, которые невозможно эффективно реализовать, используя текущее OpenCV API.

- Требования к претенденту**
- C++
 - Программирование под Linux*
- Знание OpenCV приветствуется.

Задача 20. Улучшение метрик качества VPU плагинов Inference Engine

Описание Intel разрабатывает фреймворк OpenVINO™, решающий задачу эффективного запуска нейронных сетей на Intel® архитектурах. Конкурентоспособная обработка требует не только достижения наилучшей производительности, но и удовлетворения ограничениям по точности. Задача состоит в улучшении точности работы VPU плагинов в Inference Engine: анализ текущего состояния, расширение тестового покрытия, разработка решений выявленных проблем.

- Требования к претенденту**
- Уверенные знания C++ / Python
 - Понимание основных алгоритмов и структур данных
 - Опыт работы с ОС Linux*

Будет плюсом:

- Понимание алгоритмов компьютерного зрения, машинного или глубокого обучения
- Уверенный английский язык (письменный)

Задача 21. Улучшение алгоритмов детектирования QR-кодов.

Описание Одной из достаточно востребованных задач классического компьютерного зрения остается задача расшифровки QR-кодов. На базе библиотеки OpenCV реализованы все этапы декодирования - от входного изображения до выходного текста.

Требуется решить задачи связанные со следующими этапами:

- проведение сравнительного анализа с другими библиотеками
- алгоритмическая оптимизация одного из базовых алгоритмов OpenCV
- добавление новой функциональности в этап детектирования

- Требования к претенденту**
- C++ и понимание основ ООП.
 - знание базовых алгоритмов и структур данных, линейная алгебра.

Приветствуется:

- знание базовых алгоритмов компьютерного зрения.
- опыт работы с библиотекой OpenCV.

Задача 22. Подготовка обучающих примеров по компьютерному зрению для freeCodeCamp.org с помощью OpenCV.js

Описание freeCodeCamp.org - самый популярный репозиторий на GitHub: ~300K звезд. Проект посвящен интерактивному обучению JavaScript.

OpenCV.js - это часть функционала библиотеки OpenCV, который скомпилирован в JavaScript код, что позволяет запускать существующие алгоритмы в браузере. Требуется подготовить серию обучающих примеров на основе OpenCV.js про обработку изображений и компьютерное зрение с дальнейшей интеграцией в проект freeCodeCamp.org.

Требования к претенденту Опыт решения задач обработки изображений: структура хранения, цветовые пространства, конверсия цветовых пространств, фильтрация изображений

Задача 23. Портинг алгоритмов компьютерного зрения на OpenCV Graph API

Описание Graph API (G-API) - одно из главных новшеств OpenCV 4.0, модуль, позволяющий представлять алгоритмы компьютерного зрения в виде вычислительного графа и обеспечивающий поддержку гетерогенных (на разных видах устройств) вычислений.

Для дальнейшего его развития ставится задача переноса ряда имеющихся решений компьютерного зрения на данный API, что также поможет оценке его производительности и выявлению возможных проблем.

Задача подразумевает создание алгоритма ретуширования лица (с применением как Deep Learning, так и традиционных методов компьютерного зрения), реализацию недостающих функций для G-API, анализ производительности полученного решения на доступных платформах с использованием инструментов Intel, добавление в OpenCV нового примера на основе разработанного алгоритма.

- Требования к претенденту**
- Уверенное знание C++ (включая C++11, шаблоны)
 - Навыки работы с git, cmake, отладчиками

Желательно:

- Знание основ OpenCV или обработки изображений
- Уверенный английский язык (письменный)

Задача 24. Система интеграции для открытого зоопарка моделей

Описание Задача состоит в расширении и улучшении системы непрерывной интеграции и валидации зоопарка моделей, включая поддержку изменений, предлагаемых сообществом для моделей, загрузчика моделей и демонстрационных приложений.

- Требования к претенденту**
- Уверенное знания C++ / Python
 - Понимание процессов и инструментов, применяемых в производстве ПО.
 - Опыт работы с GitHub и Jenkins приветствуется.

Задача 25. Внедрение алгоритмов компьютерного зрения в приложение

Описание Требуется внедрить несколько алгоритмов компьютерного зрения (детекторы объектов и их атрибутов, базирующиеся на нейронных сетях) в конвейер обработки данных приложения. Необходимо предусмотреть задание параметров алгоритмов через графический интерфейс пользователя и визуализацию обнаруженных объектов и их признаков.

- Требования к претенденту**
- Хорошее знание C++
 - Опыт разработки приложений с графическим интерфейсом для ОС Windows*/Linux*.
- Предпочтительно знакомство с системой контроля версий GitHub.

Задача 26. Портинг демо из продукта OpenVINO™ toolkit с языка C++ на Python. Создание новых демо

Описание В продукте существует множество демо, реализованных на C++: https://github.com/opencv/open_model_zoo/. Требуется портировать часть из них на Python.

- Требования к претенденту**
- Хорошие знания C++ и Python
 - Опыт разработки приложений под Windows*/Linux*
 - Умение работать с GitHub.

Задача 27. Исследование и реализация поддержки новых топологий нейронных сетей в OpenVINO™

Описание Задача состоит в изучении и реализации поддержки новых топологий нейронных сетей в OpenVINO™.

- Требования к претенденту**
- C++, Python
 - Понимание методов глубокого обучения и существующих топологий нейронных сетей.

Задача 28. Оптимизация алгоритма машинного обучения под GPU

Описание Требуется исследовать возможность эффективной реализации одного из алгоритмов машинного обучения для GPU и реализовать прототип алгоритма с использованием новой технологии для гетерогенного программирования.

- Требования к претенденту**
- C/C++
 - Хороший математический аппарат
 - Опыт использования OpenCL™ или CUDA

Задача 29. Разработка инструмента для исследования HPC приложений с использованием Intel® Advisor

Описание Intel® Advisor – инструмент для создания и оптимизации высокопроизводительных приложений для современных вычислительных архитектур, входящий в состав Intel® Parallel Studio. В рамках задачи необходимо разработать инструмент, проверяющий полноту покрытия данными HPC-приложений на современном аппаратном оборудовании Intel, исследовать современные HPC-приложения.

Требования к претенденту

- Основы работы на Linux*
- Python.

Знание сборки HPC приложений будет дополнительным преимуществом кандидата.

Задача 30. Автоматизация методологии характеристики MPI-приложения на основе инструментов Intel

Описание Задача нацелена на работу с MPI-статистикой. Требуется:

- Научить объекты, хранящие MPI-статистику сравниваться между собой.
- разработать визуальное представление результатов сравнение (HTML-table, Google-charts)
- Проанализировать реальный кейс проблемы с MPI часть в HPC приложении, повлиять на Intel® MPI Library.
- Предложить концепцию «сравниваемой статистики» в продукт APS (Intel® VTune™ Amplifier – Application Performance Snapshot)

Требования к претенденту Опытный пользователь Unix, базовое понимание MPI, Bash, Python/Django, базовое знание SQL.

Задача 31. Оптимизация параллельных алгоритмов стандартной библиотеки C++ для гетерогенных архитектур

Описание Задача заключается в оптимизации параллельных алгоритмов стандартной библиотеки C++ для [гетерогенных вычислительных систем](#).

Требования к претенденту

- Хорошее знание C++ (желательно C++11/14),
- шаблонное программирование,
- структуры данных и алгоритмы,
- общее представление о параллелизме;
- желателен опыт работы с OpenCL™/CUDA

Задача 32. Сравнительный анализ компиляторов на процессорах Intel

Описание Требуется исследовать эффективность и производительность различных компиляторов на процессорах Intel. Используя Python API для Intel® Advisor, необходимо обнаружить места, где кодогенерация icc оказывается не максимально эффективной для широкого набора приложений из мира высокопроизводительных вычислений.

Требования к претенденту

- Python
- Опыт работы с компиляторами GCC/ICC/LLVM
- Понимание принципов векторизации

Задача 33. Миграция и рефакторинг систем автоматического построения сэмплов и запуска тестов OpenVX

Описание OpenVX – набор инструментов для эффективного решения задач глубокого обучения и компьютерного зрения на архитектурах Intel. Реализация стандарта OpenVX – один из основных компонентов OpenVINO™, который используется всемирно известными компаниями. Разные части OpenVX исторически разрабатывались разными командами, использовавшими разные инфраструктурные решения. Пришло время объединить их - выполнить с помощью системы автоматической сборки и тестирования конфигурацию системы непрерывной интеграции с нуля. Используемые технологии – cmake и groovy.

Требования к претенденту Обязательный:

- Python

- Опыт использования git
- Является преимуществом:
- CMake
 - Groovy
 - Jenkins
 - другие скриптовые языки, например, bash

Задача 34. Профилировка Spark-приложений на Hadoop-кластере с помощью Intel® Vtune™ Amplifier

- Описание** Необходимо познакомиться с запуском Spark-приложения на Hadoop-кластере, определить инструменты, которые используются пользователями для оценки производительности Spark-приложения, а также с анализатором производительности Intel® VTune™ Amplifier, включая построение продукта из исходного кода, реализовать профилировку Spark-приложения в Intel® VTune™ Amplifier и подготовить статью, описывающую новый функционал.
- Требования к претенденту**
- C++
 - ООП

Задача 35. Анализ производительности параллельных алгоритмов стандартной библиотеки C++ (Parallel STL)

- Описание** Parallel STL (доступная в версии c++17) - это параллельные версии алгоритмов стандартной библиотеки шаблонов. Необходимо сделать анализ производительности параллельной / векторной реализации алгоритмов Parallel STL, базирующихся на библиотеке Intel® TBB и возможностях векторизации компилятора Intel.
- Требования к претенденту**
- C++11
 - STL
 - Обобщенное программирование

Задача 36. Разработка и оптимизация инструментов и алгоритмов в OpenVINO™

- Описание** Задача состоит в выполнении оптимизации существующих и разработки новых инструментов и алгоритмов OpenVINO™.
- Требования к претенденту**
- C++,
 - Python
 - Понимание методов глубокого обучения и существующих топологий нейронных сетей.

Задача 37. Создание основанного на веб-технологиях помощника для визуализации и моделирования соответствия ядер CPU и процессов MPI-приложения

- Описание** Требуется создать наглядный и удобный для пользователя инструмента, который в дополнение к основной документации поможет смоделировать и показать влияние тонких настроек библиотеки на логику раскладки процессов по ядрам всех CPU на узле в MPI-приложении (т.е. пиннинг). Для эффективного использования ресурсов крайне важно учитывать топологию CPU из-за эффектов от шаблона межпроцессного взаимодействия и балансировки нагрузки. Пиннинг является одной из самых сложных для пользователя частей документации библиотеки Intel® MPI. В процессе стажировки вы сможете изучить тонкости топологий CPU и работы с ними в MPI-приложениях, а так же приобретете практику в веб-программировании.
- Требования к претенденту**
- Опыт с HTML и JavaScript
 - Знание MPI стандарта
 - Понимание архитектуры CPU

Задача 38. Разработка эффективных алгоритмов для обработки естественных языков (NLP)

- Описание** Обработка естественных языков (NLP) как одно из направлений искусственного интеллекта позволяет решать такие прикладные задачи, как распознавание и синтез речи, генерация текста и другие. Задачи в области NLP могут быть решены с достаточной точностью на базе подходов машинного обучения, но при этом требуют значительных вычислительных ресурсов. Поэтому важно не только подобрать подходящий алгоритм, но и обеспечить его быструю работу на

доступном аппаратном обеспечении. В рамках проекта предложен алгоритм, решающий одну из прикладных задач в области NLP, который необходимо модифицировать для быстрой работы на высоко-параллельных вычислителях, таких как Intel® Xeon и NVidia GPU. Возможно применение как алгоритмической оптимизации, так и оптимизации под конкретные вычислители, включая использование готовых пакетов машинного обучения.

Требования к претенденту

- C++,
- Python,
- Структуры данных и алгоритмы.

Опыт применения алгоритмов машинного обучения и навыки оптимизации производительности приветствуются.