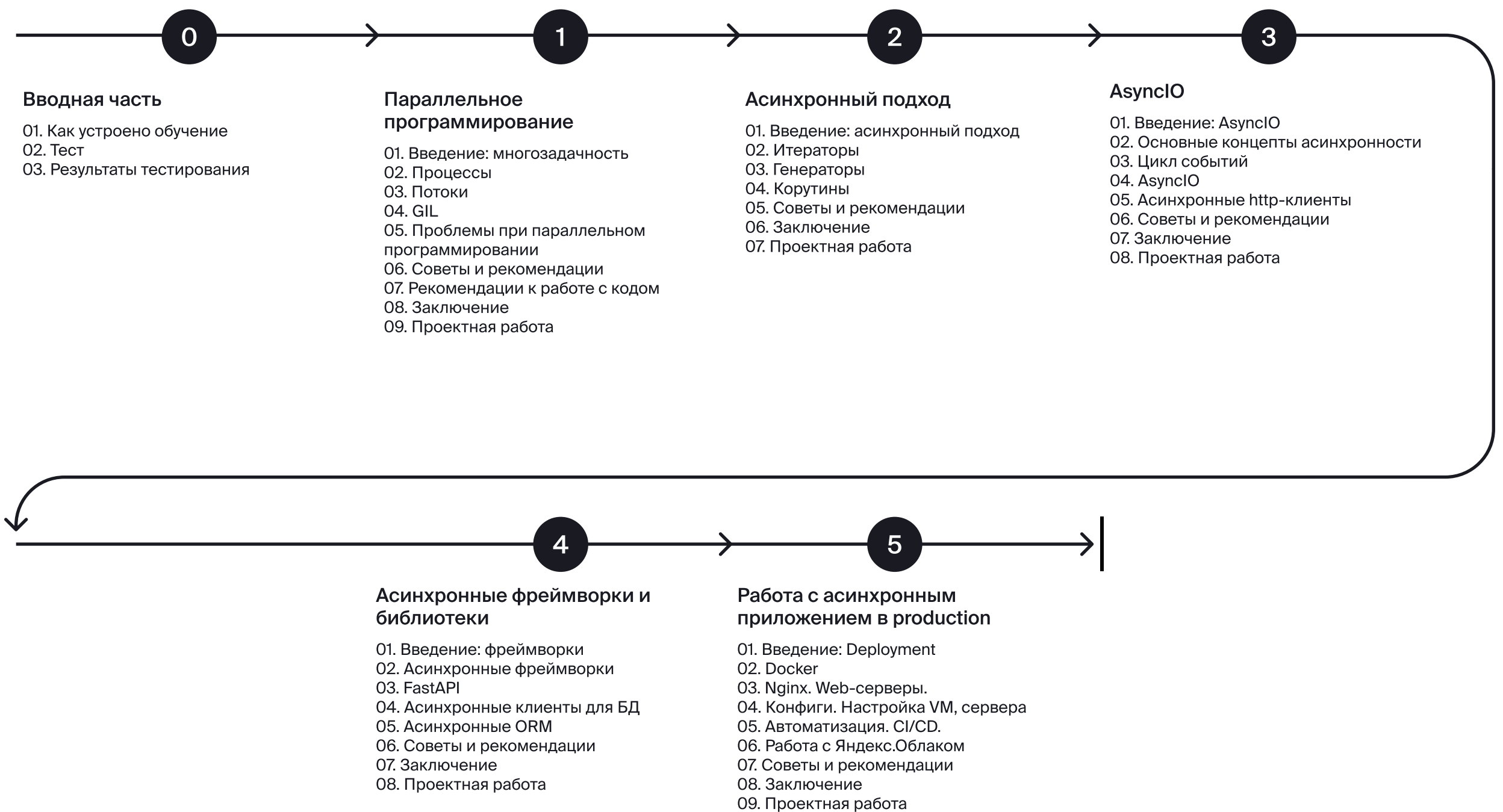


Асинхронное программирование на Python



Вступительное тестирование

3 часа

Знакомство с процессом обучения на курсе и вступительное тестирование для оценки навыков.

Спринт 1

2 недели,
30 часов,

+1 проект
в портфолио

Параллельное программирование

Знакомство с процессами и потоками: создание, завершение, поведение и проблемы, которые возникают в мультипроцессорных и многопоточных программах. Изучение GIL, понятие голодания, гонки, взаимоблокировки, приобретение навыка обходить возникающие проблемы, применение процессов и потоков при решении прикладных задач.

Технологии:

- Python: библиотеки multiprocessing, threading

Проект:

Анализ данных от API Яндекс.Погода с применением концепций параллельного программирования

Спринт 2 Асинхронный подход

2 недели,
30 часов,

+1 проект
в портфолио

Знакомство с итераторами, генераторами и корутинами, разными типами задач, разницей в утилизации ресурсов и ограничениями использования подходов. Студенты научатся создавать и использовать кастомные итераторы, генераторы и корутины.

Технологии:

- Python: итераторы, генераторы, корутины

Проект:

Проектирование и разработка таск-менеджера на основе корутин с использованием концепций асинхронного программирования

Спринт 3 AsyncIO

2 недели,
30 часов,

+1 проект
в портфолио

Изучение событийного программирования и циклов событий, знакомство с типами многозадачности. Студенты научатся использовать сокеты и применять основные классы и примитивы фреймворка asyncio для решения прикладных задач, создадут свой первый асинхронный сервер на asyncio и http-сервер для работы с запросами.

Технологии:

- Python: asyncio, aiohttp, streams, httpx

Проект:

Проектирование и разработка мессенджера (клиентской и серверной части) на основе asyncio

Спринт 4 Асинхронные библиотеки и фреймворки

2 недели,
30 часов,

+1 проект
в портфолио

Погружение в устройство фреймворков, обработку запросов, реализацию бизнес-логики. Студенты приобретут практические навыки работы с асинхронными фреймворками и библиотеками, такими как aiohttp, FastAPI, и создадут свой сервис с использованием одного из фреймворков в рамках проектного задания.

Технологии:

- Python: aiohttp, FastAPI, Docker, pydantic
- PostgreSQL, Redis, ORM, Tortoise, aiopg, asyncpg

Проект:

Проектирование и разработка сервиса по генерации ссылок на ресурсы и аналитика их использования

Спринт 5 Работа с асинхронным приложением в production

2 недели,
30 часов,

+1 проект
в портфолио

Студенты научатся организовывать инфраструктуру с применением Docker и nginx, познакомятся с популярными веб-серверами для FastAPI, освоят работу с конфигами с использованием pydantic и выполнят практическое задание по созданию сервиса, который работает с БД, деплоится в облако и подвергается нагрузке

Технологии:

- Python: FastAPI, Daphne, Hypercorn, Uvicorn, pydantic
- Docker, Docker Compose, Nginx, PostgreSQL, Redis, Yandex.Cloud, Github Actions

Проект:

Проектирование и разработка сервиса по загрузке, хранению и выдаче файлов; развертывание сервисов на облачной платформе Yandex.Cloud

Вебинары

Наставник проводит один вебинар в спринт — итого 5 вебинаров.