

Отчёт по индивидуальному проекту. Этап 5

Дисциплина “Операционные системы”

Батова Ирина Сергеевна, НММбд-01-22

06 мая 2023

Российский университет дружбы народов, Москва, Россия

Вводная часть

Целью пятого этапа индивидуального проекта является добавление к сайту всех остальных элементов, а также публикация новых постов.

- Сделать записи для персональных проектов
- Сделать пост по прошедшей неделе
- Добавить пост на тему “Языки научного программирования”

Основная часть

Создание каталогов для проектов

- Переходим в каталог `'~/work/blog/content/project'`
- Создаем по каталогу для каждого из проектов



cripts



ports

Первый проект

- Переходим в каталог 'cripts'
- Заменяем в этом каталоге картинку на подходящую по тематике



featured.jpg



index.md

- В начале файла редактируем название проекта, краткое содержание и дату

```
1
2 title: Ассиметричное шифрования
3 summary: Методы криптования на основе открытого ключа - схема реализации, преимущества и недостатки, самые известные системы
4 tags:
5   - Deep Learning
6 date: '2023-05-05T00:00:00Z'
7
8 # Optional external URL for project (replaces project detail page).
9 external_link: ''
10
11 image:
12   caption: Photo by rawpixel on Unsplash
13   focal_point: Smart
14
15
```


- В конец файла вводим текст проекта

```
21 # Slides (optional).
22 # Associate this project with Markdown slides.
23 # Simply enter your slide deck's filename without extension.
24 # E.g. `slides = "example-slides"` references `content/slides/example-slides.md`.
25 # Otherwise, set `slides = ""`.
26 slides: example
27 ---
```

28

29 При шифровании с открытым ключом каждым из собеседников создается пара ключей – открытый ключ и закрытый ключ. Эти ключи связаны друг с другом, поэтому расшифровать сообщения, зашифрованное с помощью открытого ключа, можно только с помощью соответствующего закрытого ключа. Важным моментом является то, что при создании ключей возможна генерация очень больших чисел таким образом, что при знании открытого ключа нельзя вычислить закрытый ключ никаким образом. Таким образом, обеспечение надежного шифрования сообщений базируется на знании закрытого ключа исключительно его владельцем. В основе идеи асимметричного шифрования лежит сложный математический вопрос, позволяющий строить односторонние функции. По известному x легко определяется функция $y = f(x)$. Но при этом по известному значению y очень сложно вычислить x . В качестве открытого ключа используется y , а в качестве закрытого x . Для всех методов шифрования с открытым ключом математически строго не доказано отсутствие других методов вычисления закрытого ключа методом перебора. Криптосистемы с открытыми ключами различаются видом односторонних функций.

31

32 При необходимости передачи информации от собеседника к собеседнику действует единая схема реализации асимметричного шифрования. Для начала первый собеседник выбирает алгоритм и пару открытого и закрытого ключей. Открытый ключ посылается второму собеседнику по открытому, незащищенному каналу. После этого второй собеседник шифрует информацию с помощью присланного открытого ключа и передает полученный шифр обратно первому собеседнику, который расшифровывает сообщение с помощью созданного в начале закрытого ключа. При этом, если необходимо наладить канал связи в обе стороны, то операции до расшифровки сообщений делают оба собеседника. Таким образом, каждый будет знать свои закрытый, открытый ключи и открытый ключ собеседника.

Второй проект

- Переходим в каталог 'ports'
- Заменяем в этом каталоге картинку на подходящую по тематике



featured.jpg



index.md

- В начале файла редактируем название проекта, краткое содержание и дату

```
1 ---
2 title: Параллельные и последовательные порты
3 summary: Особенности использования компьютерных портов - точки соединения между компьютером и периферийными устройствами
4 tags:
5   - Deep Learning
6 date: '2023-05-05T00:00:00Z'
7
8 # Optional external URL for project (replaces project detail page).
9 external_link: ''
10
11 image:
12   caption: Photo by rawpixel on Unsplash
13   focal_point: Smart
14 ..
```

- В конец файла вводим текст проекта

```
20
21 # Slides (optional).
22 # Associate this project with Markdown slides.
23 # Simply enter your slide deck's filename without extension.
24 # E.g. 'slides = "example-slides"' references 'content/slides/example-slides.md'.
25 # Otherwise, set 'slides = ""'.
26 slides: example
27 ---
28
```

29 Для связи с периферийными устройствами к шине компьютера подключены одна или несколько микросхем контроллера ввода-вывода. Основная функция порта – выступать в качестве точки подключения, в которую подключают кабель от периферийного устройства, и обеспечить передачу данных от одного устройства к другому. Обычно именно сам конец разъема, находящегося на материнской плате, называют портом. В любом компьютере обязательно есть минимальный набор портов, без которых функционирование невозможно – это порты для подключения монитора, мыши, клавиатуры, разъем для сетевого подключения, универсальный USB порт и аудиоразъем звуковой платы. При необходимости количество компьютерных портов можно увеличить с помощью плат расширения, подключаемых к основной плате.

30
31 Порты можно разделить на два типа – последовательные и параллельные. Делят их в зависимости от типа, используемого для связи.

32
33 Последовательные порты (также называется COM-портом) – это порты, через которые периферийные устройства подключаются с использованием последовательного протокола. Другими словами, информация через них передается по одному биту, бит за битом. Название «последовательный порт» закрепилось за портом, имеющим стандарт RS-232C, поэтому, хотя другие интерфейсы (например, USB) также используют последовательный способ обмена, их не относят к последовательным портам. Последовательные разъемы обычно имеют от 9 до 25 контактов, а персональный компьютер обычно имеет от одного до четырех последовательных портов. Последовательные порты, как правило, встраиваются в материнскую плату.

34
35 Особенностью последовательных портов по сравнению с другими «последовательными» технологиями (например, USB или Ethernet) является присутствие временных требований только между битами одного байта (скорость передачи это и есть временная пауза между ними). Какие-либо временные требования между 2 байтами отсутствуют.

36

Размещение поста о прошедшей неделе

- Редактируем название и дату

```
4 title: "Неделя с 01.05 по 07.05"  
5 subtitle: ""  
6 summary: ""  
7 authors: [admin]  
8 tags: [Academic]  
9 categories: [Demo]  
0 date: 2023-05-05T13:08:21+03:00  
1 lastmod: 2023-05-05T13:08:21+03:00  
2 featured: false  
3 draft: false
```

Размещение поста о прошедшей неделе

- В конец файла вводим текст поста

```
23 # Projects (optional).
24 # Associate this post with one or more of your projects.
25 # Simply enter your project's folder or file name without extension.
26 # E.g. `projects = ["internal-project"]` references `content/project/deep-learning/index.md`.
27 # Otherwise, set `projects = []`.
28 projects: []
29 ---
30
```

31 На этой неделе, в связи с приближением сессии, было много учебы и работы. Я продолжила работать над сайтом, выполнила предпоследнюю лабораторную работу.

Размещение поста о языках научного программирования

- Редактируем название и дату

```
1 ---
2 # Documentation: https://wowchemy.com/docs/managing-content/
3
4 title: "Языки научного программирования"
5 subtitle: ""
6 summary: ""
7 authors: [admin]
8 tags: [Academic]
9 categories: [Demo]
10 date: 2023-05-05T13:09:21+03:00
11 lastmod: 2023-05-05T13:09:21+03:00
12 featured: false
13 draft: false
```

Размещение поста о языках научного программирования

- В конец файла вводим текст поста

```
--
23 # Projects (optional).
24 # Associate this post with one or more of your projects.
25 # Simply enter your project's folder or file name without extension.
26 # E.g. 'projects = ["internal-project"]' references 'content/project/deep-learning/index.md'.
27 # Otherwise, set 'projects = []'.
28 projects: []
29 ---
30
31 Как известно, в мире множество языков научного программирования. В этом посте немного поговорим о самых известных и используемых языках в разных сферах науки.
32
33 К одному из важных языков относится Bash – обновленная оболочка Unix, впервые выпущенная в конце 80-х годов. На Bash обычно пишутся сценарии конфигурации сервера. Области применения:
    администрирование сервера, инструменты интерфейса командной строки, Docker, виртуализация.
34
35 К одному из старейших, но используемых до сих пор языков, относится COBOL. Он был создан Греис Хоппер в конце 1950-х годов и предназначался для стандартизации разработки крупномасштабных бизнес-
    приложений на компьютерах. Через системы COBOL проходят невероятные 95% транзакций АТМ, причем многие из этих устройств работают круглосуточно 365 дней в году. Области применения: банковское дело/
    финансы, международные корпорации, правительства.
36
37 Самыми мощными и бистродействующими языками считаются C/C++. C был создан в начале 1970-х годов Деннисом Ритчи, C++ в 1985 году разработал Бьёрн Страуструп. C и C++ хорошо подходят для
    требовательных приложений, которым важны низкоуровневый доступ к оборудованию и производительность. Область применения: разработка ОС/SDK, встроенные устройства, игровые движки, наука.
38
```


Вывод

В ходе пятого этапа индивидуального проекта я добавила к сайту все остальные элементы, а также опубликовала новые посты.