

Управление содержанием проекта - окончание

Лектор – Скороход С. В.

Подтверждение содержания

- Это формальное принятие участником проекта завершеного содержания проекта и относящихся к нему конечных результатов.
- Верификация содержания проекта включает в себя проверку результатов для определения полной готовности каждого из них.
- Важнейшими составляющими подтверждения содержания программного обеспечения являются его тестирование, верификация и валидация.

Понятие тестирования

- **Тестирование программного обеспечения (Software Testing)** - проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом
- **В более широком смысле, тестирование** - это одна из техник контроля качества, включающая в себя действия по планированию работ (**Test Management**), проектированию тестов (**Test Design**), выполнению тестирования (**Test Execution**) и анализу полученных результатов (**Test Analysis**).
- **Верификация (Verification)** - это процесс оценки системы или её компонентов с целью определения удовлетворяют ли результаты текущего этапа разработки условиям, сформированным в начале этого этапа. Т.е. выполняются ли наши цели, сроки, задачи по разработке проекта, определенные в начале текущей фазы.
- **Валидация (Validation)** - это определение соответствия разрабатываемого ПО ожиданиям и потребностям пользователя, требованиям к системе

Верификация ПО

- Верификация является более общим понятием, чем тестирование.
- Ее цель – достижение определенных гарантий того, что верифицируемый объект (ПО) полностью соответствует требованиям, полностью реализован и может удовлетворить все критерии проектной спецификации и оговорённые стандарты качества.
- Любой процесс верификации включает в себя инспекции, проверку программного кода, анализ итогов тестирования, формирование анализов и отчетной документации, где будут детально расписаны все проблемы определенного проекта.
- В сфере тестирования ПО есть общепринятый постулат о том, что процесс тестирования является неотъемлемой частью верификации.
- Пример типичной верификации: процесс проведения тестирования системного компонента. Опираясь на определенные требования, мы проводим процесс тестирования и документируем, соблюдены ли требования. В конце верификации мы получаем ответ на вопрос: «Отвечает ли продукт установленным требованиям?».

Валидация ПО

- В работе с программным обеспечением валидацией считается процесс, целью которого является доказательство того, что во время функционирования системы достигаются результаты, изначально запланированные для ее непосредственного использования.
- Другими словами, валидация – это проверка соответствия созданной системы тому ожиданию, которое озвучил клиент.
- Валидация отвечает на вопрос: всецело ли данная система соответствует тем ожиданиям клиента, которые были ранее озвучены?

Отличие верификации и валидации ПО

Верификация	Валидация
По факту отвечает на вопрос, правильно ли создается и тестируется ПО и все ли требования учитываются при этом.	Отвечает на вопрос, создается ли продукт правильно с точки зрения ожиданий клиента.
В процессе верификации убеждаемся в том, что весь созданный функционал приложения работает корректно и логически верно.	При процессе валидации убеждаемся в том, что продукт полностью соответствует поведению, которое от него ожидается и то, что клиент знает о наличии подобного функционала.
В структуру верификации входят такие компоненты, как сверка установленным требованиям, технической документации и корректное выполнения программного кода на любом этапе создания и тестирования ПО.	Валидация, по своей сути, в большей степени включает в себя общую оценку ПО и может основываться исключительно на субъективном мнении касательно правильности работы приложения или его компонентов.

Пример верификации и валидации ПО

- Предположим, что у нас есть сайт.
- Для входа на страницу веб-сайта, пользователю необходимо выполнить регистрацию или же войти в систему под своим аккаунтом.
- Имеется определенная форма с полями, которые необходимо заполнить.

Login
/
Registration

Name

Adress

Password

Submit

Пример верификации и валидации ПО

Верификация

- Проверим наличие полей. Все поля должны быть валидными и соответствовать требованиям спецификации.
- Их количество, отображение и особенности определяются дизайнерами, которые создают макеты. Необходимые данные вносятся в техническое задание, а в случае отсутствия такового – необходимо иметь доступы к созданным макетам.
- При выполнении верификации необходимо выяснить, что **все поля изначально рабочие, и в них можно занести данные, согласно отображенным обозначениям и наименованиям.**

Валидация

- В процессе валидации проверяются вводимые в поля данные, а также их соответствие утвержденной спецификации.

Пример верификации и валидации ПО

- Рассмотрим пример на основе формы входа/регистрации в рамках социальной сети Facebook.
- У нас есть спецификация и наличие полей на целевой странице ей всецело соответствует. Исходя из этого факта, верификация уже прошла успешно.

facebook

Email or Phone Password Log In
Forgotten account?

Facebook helps you connect and share with the people in your life.

Create an account
It's free and always will be.

First name Surname

Mobile number or email address

New password

Birthday
19 Apr 1994 Why do I need to provide my date of birth?

☐ Female ☐ Male


By clicking Sign Up, you agree to our Terms, Data Policy and Cookie Policy. You may receive SMS notifications from us and can opt out at any time.

Sign Up

Create a Page for a celebrity, band or business.

Пример верификации и валидации ПО

- Валидация. Чтобы не затягивать процесс, изначально введем неверную информацию в поля ввода данных.
- Введенная информация не прошла валидацию и система не пустила нас дальше, что и нужно было протестировать. **Система отвечает ожиданиям заказчика!**



The screenshot displays the Facebook 'Create an account' page. The top navigation bar is dark blue with the Facebook logo on the left and login fields (Email or Phone, Password, Log In) on the right. Below the navigation bar, the main content area is light blue. On the left, there is a graphic with the text 'Facebook helps you connect and share with the people in your life.' and a map showing connections between people. On the right, the 'Create an account' section is titled 'It's free and always will be.' and contains several input fields. The 'Mobile number or email' field is highlighted with a red border and a red error message 'What's your name?'. The 'New password' field is also highlighted with a red border and a red error message. The 'Birthday' section shows a date of birth (19 Apr 1994) and gender options (Female, Male). At the bottom, there is a green 'Sign Up' button and a link to 'Create a Page for a celebrity, band or business.'

facebook

Email or Phone Password Log In

Forgotten account?

Facebook helps you connect and share with the people in your life.

Create an account

It's free and always will be.

loreem Surname

Mobile number or email What's your name?

New password

Birthday

19 Apr 1994 Why do I need to provide my date of birth?

Female Male

By clicking Sign Up, you agree to our Terms, Data Policy and Cookie Policy. You may receive SMS notifications from us and can opt out at any time.

Sign Up

Create a Page for a celebrity, band or business.

Результаты верификации и валидации ПО

Результаты верификации и валидации фиксируются в документальной форме.

Таблица 6.2 – Результаты верификации программного кода и видеок кадров

№ п.п	Нормативный документ	Проверяемые факторы и/или критерии	Проверка соответствия
Результаты верификации программного кода ППО ПЛК шкафов ПТК			
1	Техническое задание	Структурированность	соответствует
		Модульность	соответствует
		Функциональная полнота	соответствует
		Интерфейсы связи между модулями	соответствует
		Тестируемость ПО	соответствует
		Самодиагностика ПО	соответствует
		Функциональное разделение	соответствует
		Апробированность	соответствует
		Защита от искажений	соответствует
		Защита от ошибочных действий	соответствует
		Защита от несанкционированного доступа	соответствует
		Контроль достоверности	соответствует
		Сохранность информации при авариях	соответствует
		Стойкость алгоритмов к перезагрузке ПЛК	соответствует
		Стойкость алгоритмов к отказам технических средств	соответствует
Требования к кодированию	соответствует		
Модифицируемость	соответствует		

Управление сроками проекта

Лектор – Скороход С. В.

Управление сроками



Определение состава операций: методы

1. Декомпозиция.

- Разбиение проектных работ на более мелкие и более управляемые элементы, называемые плановыми операциями.
- Каждый пакет работ в ИСР разбивается на плановые операции, необходимые для получения результатов этого пакета.
- Формулировка операции должна быть максимально конкретной и указывать на ее результат.

2. Шаблоны.

- В качестве шаблона можно использовать список операций из предыдущего проекта, если имеются элементы совпадения или подобия текущего проекта с предыдущим.

3. Метод набегающей волны.

- Работа, которую надо будет выполнить в ближайшей перспективе, подробно планируется на низшем уровне ИСР, а далеко отстоящая работа планируется на сравнительно высоком уровне ИСР.
- Планирование работ на один-два ближайших периода уточняется по мере выполнения работ в текущем периоде.
- Поэтому на разных стадиях ЖЦ проекта плановые операции могут иметь разную степень конкретизации.

Определение состава операций: результаты

1. Список операций.

- Это исчерпывающий перечень, включающий в себя все плановые операции проекта.
- Описание содержания работ по каждой из них, подробное настолько, чтобы члены команды проекта понимали, какие работы необходимо провести.
- Содержание работ операции может выражаться в физических величинах. Например, количество полей в интерфейсном окне.

2. Параметры операций.

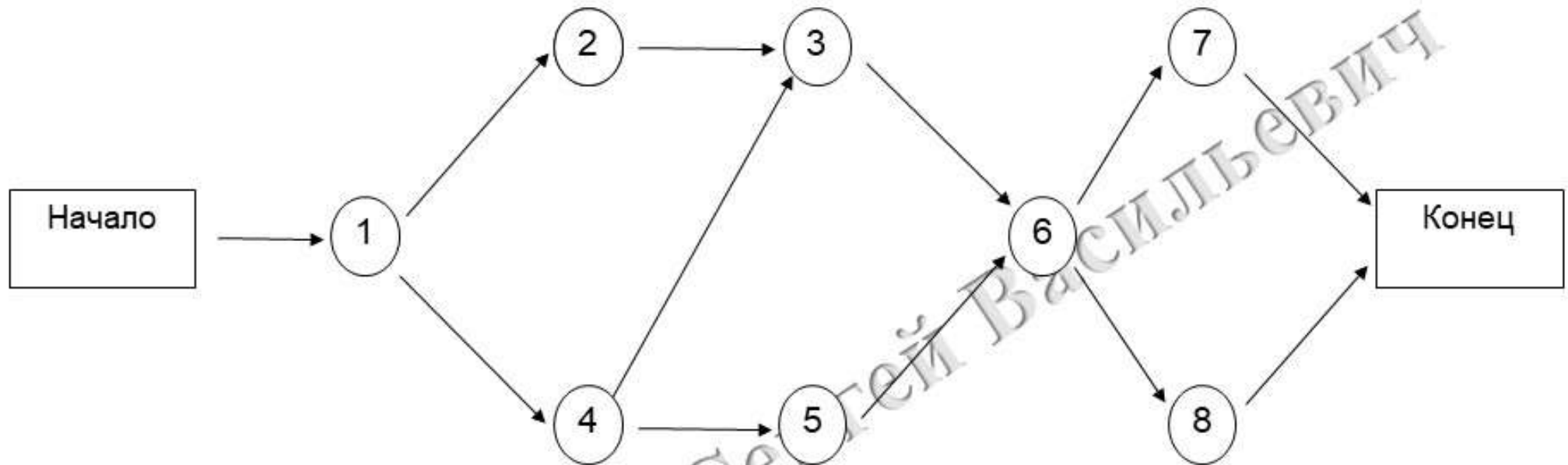
- Идентификатор операции, ее описание, перечни предшествующих и последующих операций, логические взаимосвязи, опережения и задержки, требования к ресурсам, требуемые даты, ограничения и допущения. Могут включать ответственного за выполнение работы, местоположение выполнения работ и тип планирования.

1. Список контрольных событий.

- Задаёт все контрольные события расписания, указывая при этом, является ли событие обязательным (необходимым согласно контракту) или необязательным.

Определение взаимосвязей операций

Метод предшествования (или «операции в узлах»)



Типы зависимостей

- 1. Финиш-старт.** Инициация последующей операции зависит от завершения предшествующей операции.
- 2. Финиш-финиш.** Завершение последующей операции зависит от завершения предшествующей операции.
- 3. Старт-старт.** Инициация последующей операции зависит от инициации предшествующей операции.
- 4. Старт-финиш.** Завершение последующей операции зависит от инициации предшествующей операции.

Виды взаимосвязей операций

- **Обязательные зависимости.** Это зависимости, которые являются неотъемлемым свойством выполняемой работы. Они часто подразумевают физические ограничения. Например прототип должен быть создан до того, как он будет протестирован. Обязательные зависимости часто называют **жесткой логикой**.
- **Произвольные зависимости.** Их называют предпочитаемой логикой или мягкой логикой. Они обычно устанавливаются на основе передовых методов организации работ, где желательная особая последовательность операций, несмотря на то, что имеются и другие приемлемые последовательности (методологии ЖЦ разработки ПО).
- **Внешние зависимости.** Включают взаимоотношения операций проекта с непроектными операциями. Например, в проекте по разработке программного обеспечения сроки операции тестирования могут зависеть от поставки аппаратного обеспечения сторонней организацией.

Задержки и опережения

- **Опережение** позволяет ускорить последующую операцию. Например, программист может приступить к проектированию архитектуры системы до того как системный аналитик завершит описание требований к интерфейсу. Это может быть достигнуто при помощи взаимосвязи "финиш-старт" с пятнадцатидневным опережением.
- **Задержка** управляет приостановкой последующей операции. Например, после завершения модульного тестирования перед началом интеграционного тестирования нужно провести профилактику оборудования в течение 3 рабочих дней. Используется задержка во взаимосвязи "финиш-старт" длительностью 3 дня.

Оценка ресурсов операций

- Определяет, какие ресурсы (человеческие, оборудование, материалы, финансы) будут использоваться и в каком количестве, и когда каждый из них будет доступен для выполнения проектных операций.

Результаты оценки.

- **перечень** и описание типов и количества ресурсов, необходимых для каждой плановой операции в пакете работ;
- этот перечень можно затем собрать в единое целое для определения оценочных ресурсов по каждому пакету работ в целом;
- **календарь ресурсов** проекта, который устанавливает рабочие и нерабочие дни, на которые данный ресурс может быть активным или не может быть задействован.

Оценка длительностей операций

1. Экспертная оценка.

Используется накопленный опыт и историческая информация экспертов как из предыдущих, так и из текущего проекта.

2. Оценка по аналогам.

Использование фактической длительности аналогичной предыдущей плановой операции в качестве основы для оценки длительности будущей плановой операции.

3. Параметрическая оценка.

Используется когда можно точно определить физический объем операции и производительность ресурса:

$$\text{Длительность} = \frac{\text{физический объем}}{\text{производительность ресурса}}.$$

Оценка длительностей операций

4. Оценка по трём точкам.

- **Наиболее вероятная (НВ).**

Длительность операции с учетом предварительного выделения ресурсов, их производительности, реалистичной оценки их доступности, а также задержек.

- **Оптимистичная (О).**

Длительность операции основывается на оптимистичном сценарии, описанном в наиболее вероятной оценке.

- **Пессимистичная (П).**

Длительность операции основывается на оптимистичном сценарии, описанном в наиболее вероятной оценке.

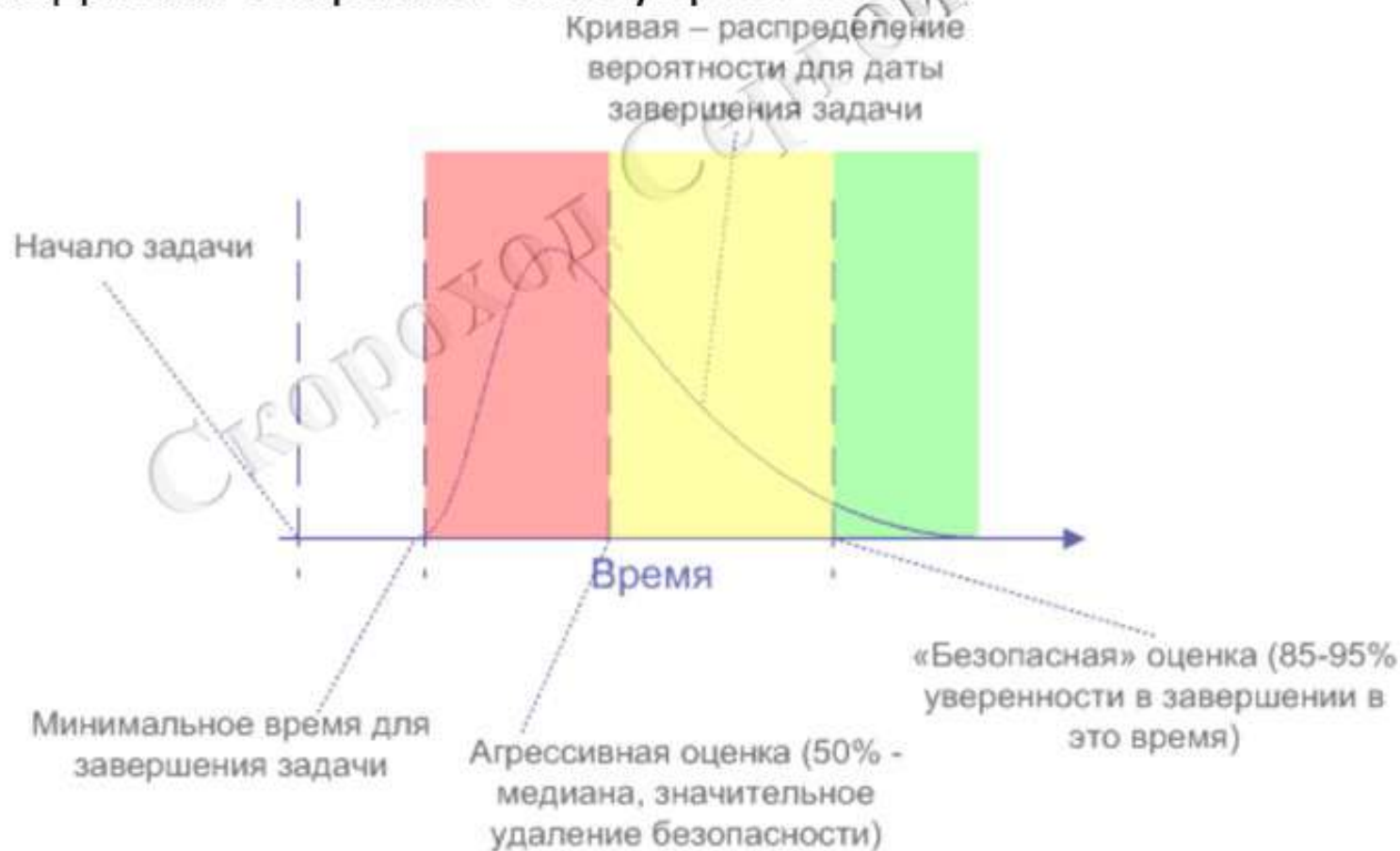
$$\text{длительность} = \frac{O + 4 * \text{НВ} + П}{6}$$

5. Анализ резервов.

Оценка длительностей операций

5. Анализ резервов.

- Добавление дополнительного времени, называемого временным резервом или буфером, в общее расписание проекта в качестве учета рисков нарушения графика.
- Резерв может добавляться для операций, фаз и проекта в целом.
- Резерв можно использовать полностью или частично, его можно впоследствии сократить или убрать вовсе.



Разработка расписания: методы

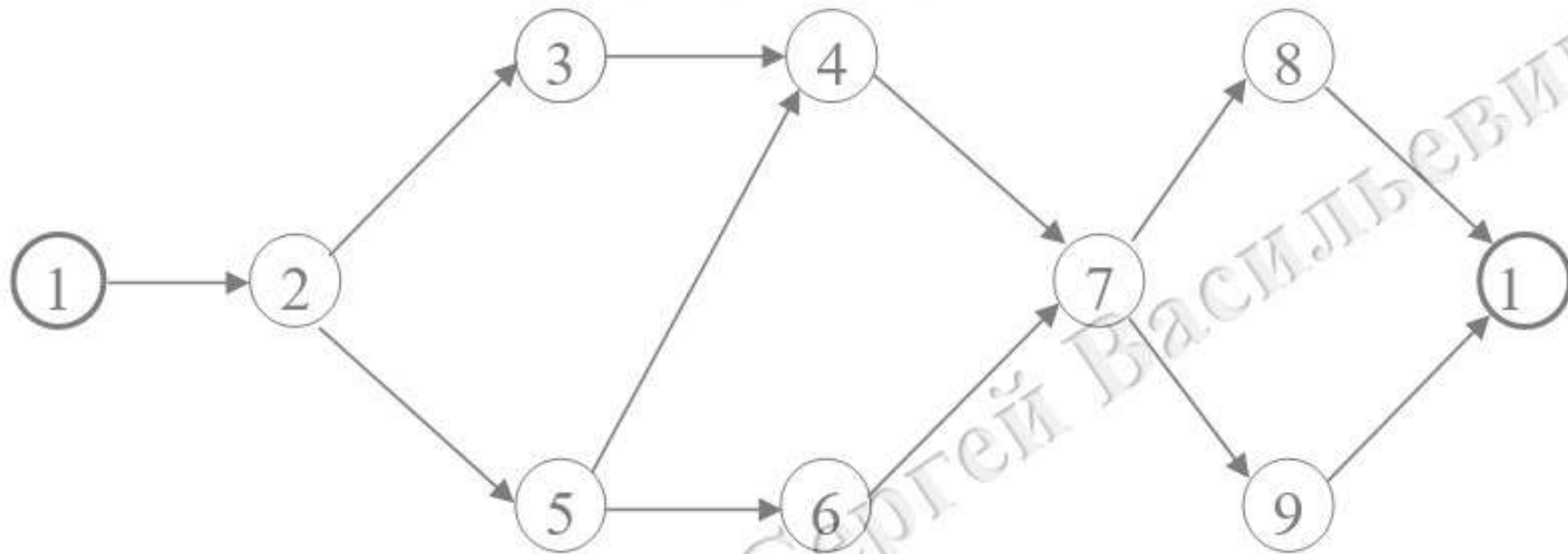
Разработка расписания проекта – это итеративный процесс, определяющий плановые даты начала и завершения операций проекта.

Методы составления расписания.

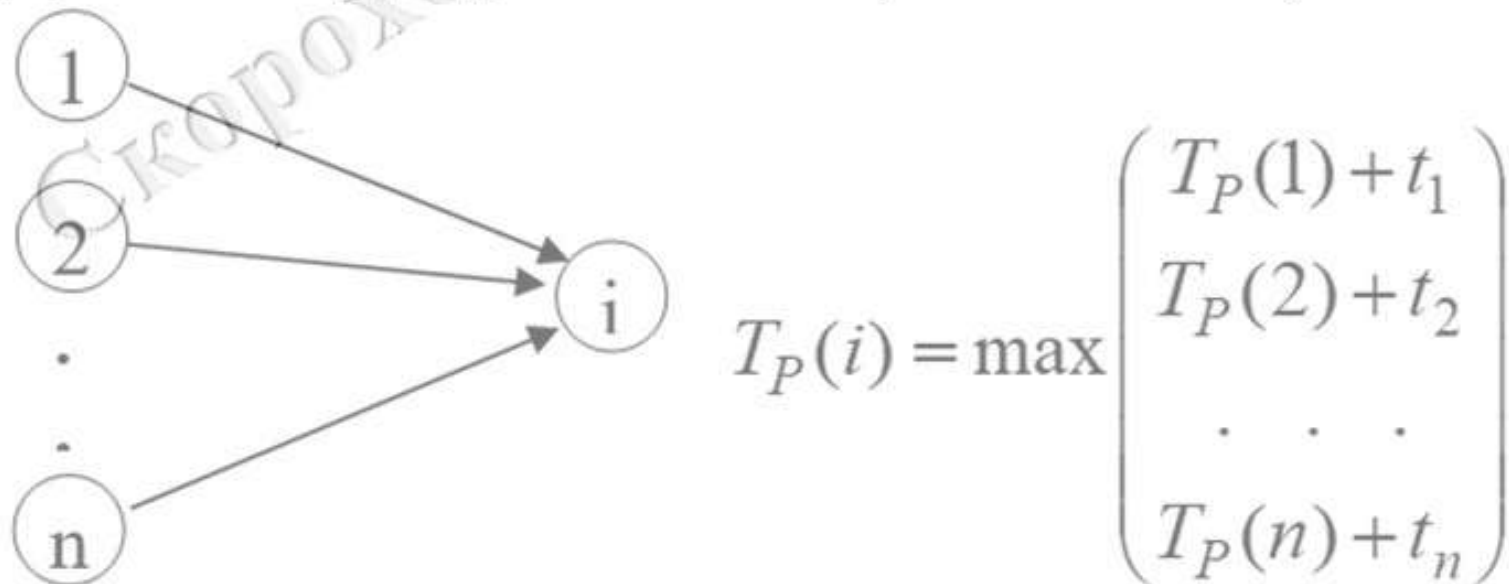
1. Метод критического пути.
2. Сжатие расписания.
 1. Сжатие.
 2. Быстрый проход.
3. Анализ возможных сценариев.
4. Выравнивание ресурсов.
5. Метод критической цепи.

Метод критического пути

1. Составить сетевой график работ проекта.

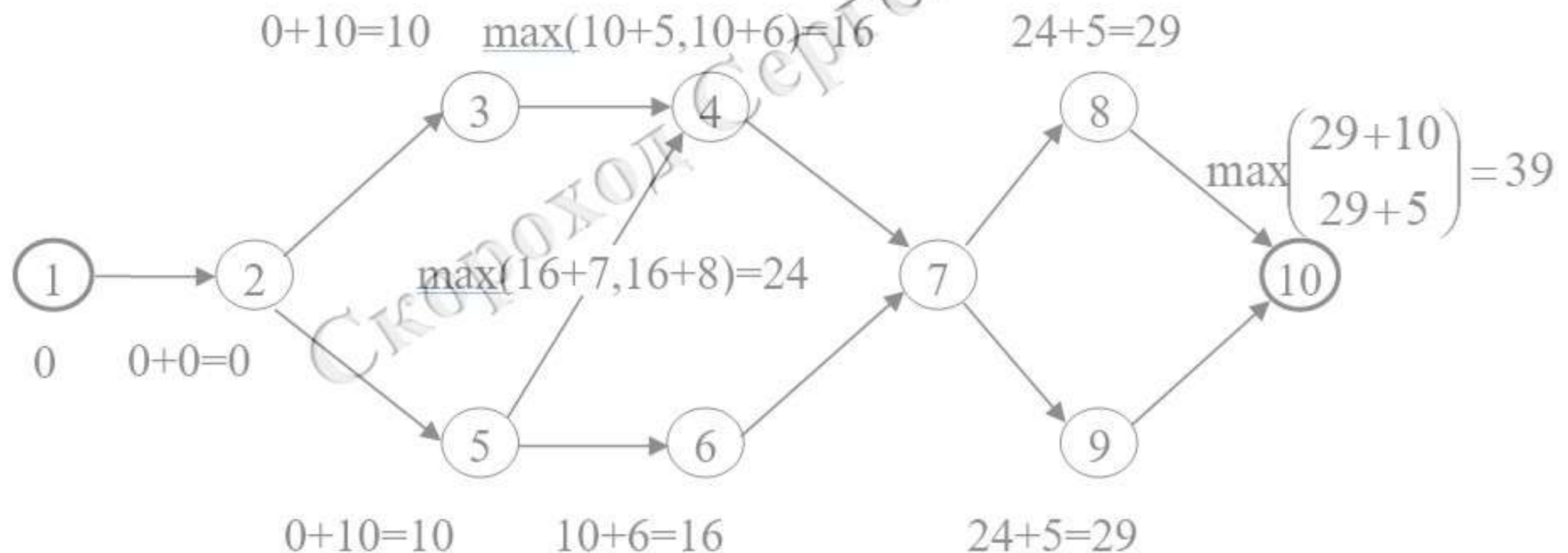


2. Вычислить раннее время начала работ (время относительно начала проекта, ранее которого работы не могут быть начаты)



Метод критического пути

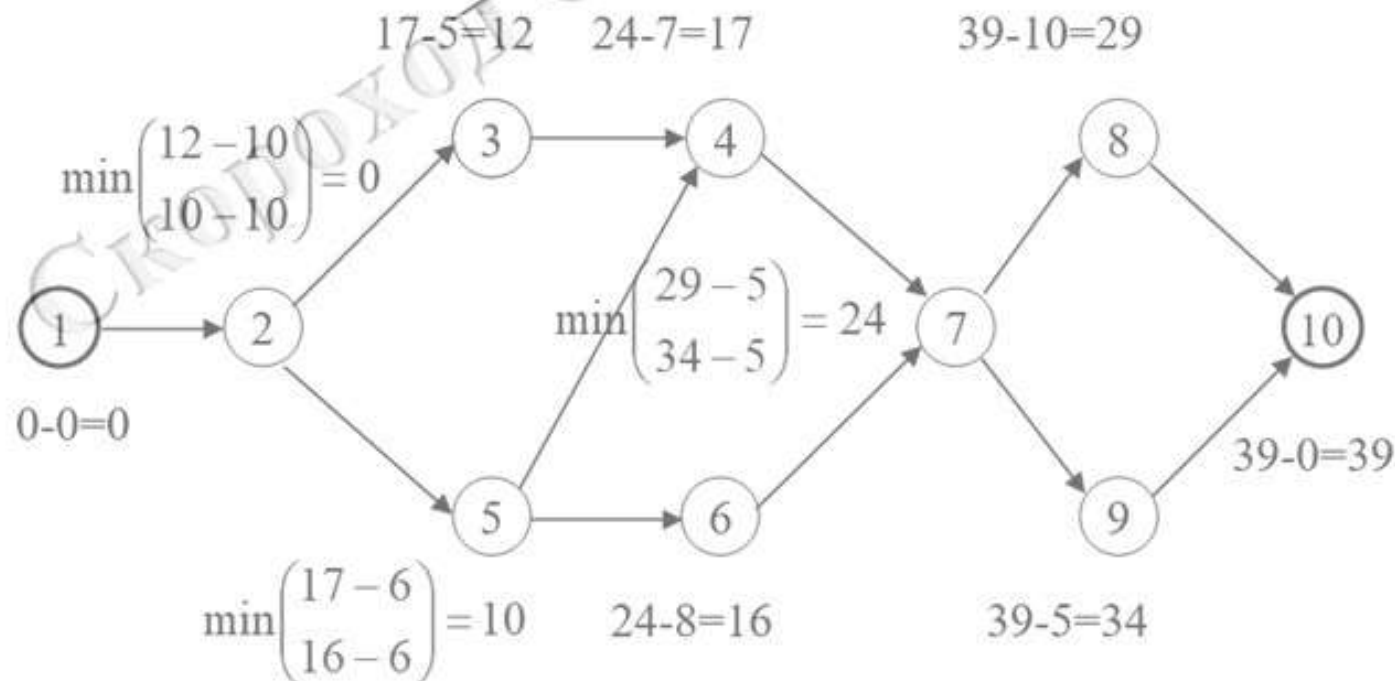
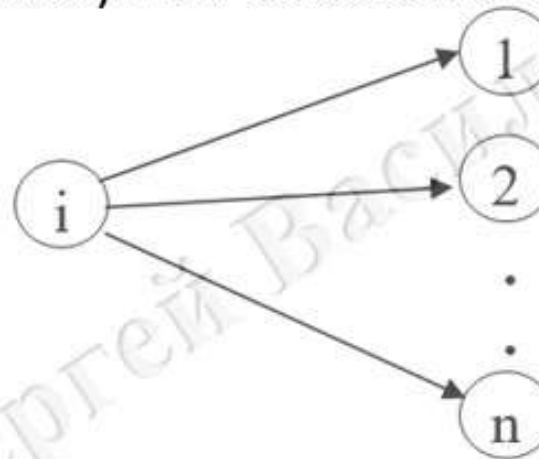
Номер работы	Длительность	Номер работы	Длительность
1	0	6	8
2	10	7	5
3	5	8	10
4	7	9	5
5	6	10	0



Метод критического пути

3. Вычислить позднее время начала работ (время относительно начала проекта, позже которого работы не могут быть начаты без увеличения времени всего проекта). Вычисление ведется от конца.

$$T_{\Pi}(i) = \min \begin{pmatrix} T_{\Pi}(1) - t_1 \\ T_{\Pi}(2) - t_2 \\ \dots \\ T_{\Pi}(n) - t_n \end{pmatrix}$$

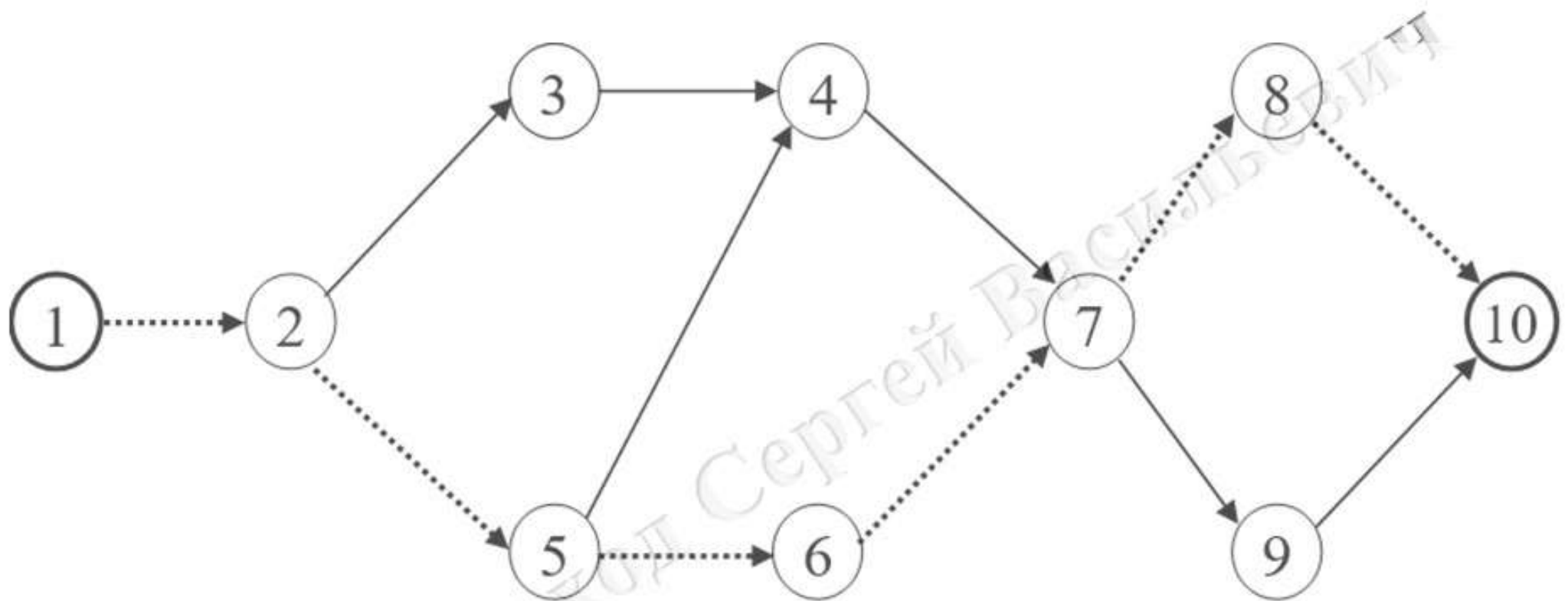


Метод критического пути

3. Вычислить резервы работ как разность позднего и раннего времени начала.
4. Работы с нулевым резервом – критические работы. Их начало нельзя задержать без удлинения всего проекта в целом.
5. Критический путь – путь от начальной до конечной работы, проходящий только через критические работы.

<i>Работа</i>	1	2	3	4	5	6	7	8	9	10
<i>Раннее время начала</i>	0	0	10	16	10	16	24	29	29	39
<i>Позднее время начала</i>	0	0	12	17	10	16	24	29	34	39
<i>Резерв времени</i>	0	0	2	1	0	0	0	0	5	0

Метод критического пути



Сжатие расписания

Сжатие расписания укорачивает расписание проекта без изменения содержания проекта, с сохранением ограничений на сроки.

Методы сжатия расписания.

- **Сжатие.** Анализ компромиссов стоимости и сроков: каким образом можно максимально сжать сроки при минимальных дополнительных затратах. Сжатие не всегда позволяет получить приемлемое решение и может привести к увеличению стоимости. Примером сжатия является увеличение числа сотрудников, занятых на длительной операции с целью уменьшения ее длительности.
- **Быстрый проход.** Фазы или операции, обычно выполняемые последовательно, проводятся параллельно. Быстрый проход может привести к доработкам и возрастанию риска. Пример. Программирование отдельных модулей до окончания формулирования требований к программному комплексу.

Анализ возможных сценариев

- Сценарии формулируются как вопросы типа "Что произойдет, если ситуация будет развиваться по сценарию 'X'?"»
- Выполняется анализ расписания, при возникновении некоторого события (риска). Например, задержка поставки основного компонента или увеличение длительности отдельных инженерных операций.
- Результаты анализа могут использоваться для оценки выполнимости расписания в условиях риска и для составления планов реагирования на риски.