

## Лабораторная работа №1

### УПРАВЛЕНИЕ ПЛАНИРОВАНИЕМ ПРОСТОГО ПРОЕКТА В СРЕДЕ *Gantt Project*

Задача управления ходом реализации проекта, распределения доступных ресурсов является очень важной для успешного выполнения проекта в заданные сроки и в рамках выделенного финансирования.

Для автоматизации заданной задачи целесообразно использовать специализированное программное обеспечение — *системы управления проектами*. Наиболее известной системой заданного класса является программа *Microsoft Project* — мощный коммерческий пакет. Неплохой бесплатной альтернативой *Microsoft Project* может стать программа *GanttProject*, которая рассматривается в рамках данной работы.

#### 1. Постановка задачи

Рассмотрим пример создания небольшого проекта. Пусть необходимо составить проект выполнения разработки компьютерной программы, состоящей из двух модулей. Собственно процесс создания такой программы содержит:

- 1) разработку алгоритма;
- 2) программирование каждого модуля;
- 3) совместную отладку модулей;
- 4) оформление программной документации (рисунок 1).

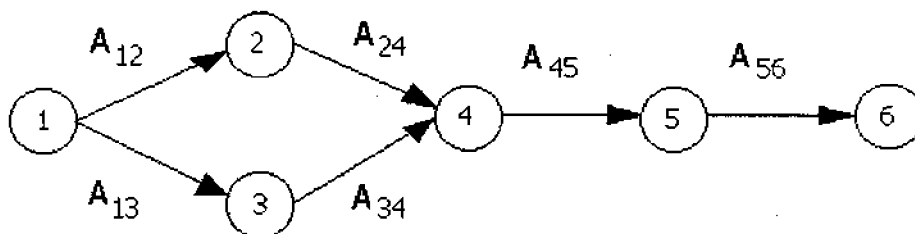


Рисунок 1. Сетевой график выполнения проекта

- $A_{12}$  - разработка алгоритма первого модуля;
- $A_{13}$  - разработка алгоритма второго модуля;
- $A_{24}$  - программирование первого модуля;
- $A_{34}$  - программирование второго модуля;
- $A_{45}$  - комплексная отладка модулей;
- $A_{56}$  - разработка программной документации.

При разработке сетевых графиков необходимо учитывать следующие правила:

1. Каждая работа на сетевом графике представляется одной и только одной дугой, то есть ни одна работа не должна появляться в графике дважды. При этом любая работа в случае необходимости может быть разбита на две или более частей, каждой из которых будет соответствовать своя дуга. Например, программирование модуля можно представить как две работы: ввод текста программы и ее отладку.

2. Ни одна пара работ не должна определяться одинаковыми начальным и конечным событиями (в противном случае две различные работы будут идентифицированы одинаково). Возможность такого неоднозначного задания работ существует в тех случаях, когда две (или более) работы могут выполняться одновременно.
3. Ни одно событие не может произойти до тех пор, пока не будут закончены все входящие в него работы. Например, комплексная отладка модулей не может быть начата до завершения программирования каждого из них, то есть событие  $A_{45}$  на рис.1 не произойдет до завершения работ  $A_{24}$  и  $A_{34}$ ;
4. Ни одна работа, выходящая из данного события, не может начинаться до тех пор, пока не произойдет данное событие; например, программирование первого модуля (работа  $A_{24}$ ) не может начаться, если не закончена разработка его алгоритма (то есть пока не произошло событие  $A_{12}$ ). Чтобы исключить неоднозначность, вводят фиктивную работу и фиктивное промежуточное событие. Затраты времени и ресурсов на выполнение фиктивной работы принимаются равными нулю.

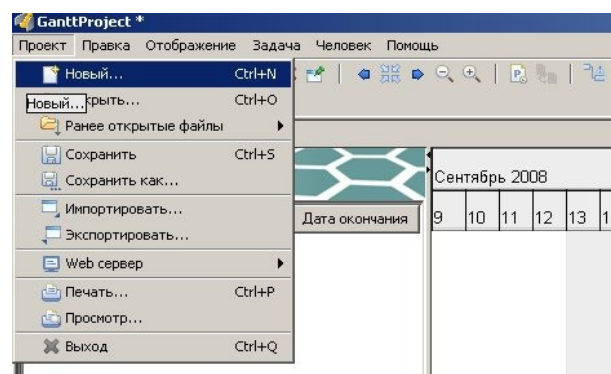
Зададим длительности перечисленных работ:

- разработка алгоритма первого модуля — 6 дней;
- разработка алгоритма второго модуля — 8 дней;
- программирование первого модуля — 9 дней;
- программирование второго модуля — 10 дней;
- комплексная отладка модулей — 4 дня;
- разработка программной документации — 5 дней.

## 2. Создание проекта и установка параметров

Для создания и инициализации проекта необходимо выполнить начальные действия:

1. Выбрать пункт главного меню *Проект -> Новый* (рисунок 2).



**Рисунок 2. Создание нового проекта**

2. Задать последовательно параметры нового проекта в полях формы параметров (рисунок 3), завершая ввод нажатием кнопки «Вперёд >».

**Новый проект** (шаг 1 из 3)

Название  
Разработка программы

Организация  
Фирма 1-Эс

Веб-страница  
www.1es.ru

Описание  
Проект выполнения разработки компьютерной программы, состоящей из двух модулей.

< Назад    Вперед >    ОК    Отмена

**Рисунок 3. Окно параметров нового проекта**

3. Выбрать предметную область проекта — «Разработка программного обеспечения» (рисунок 4). При этом с проектом будет связан стандартный набор ролей участников проекта, список которых приводится в всплывающей подсказке.

**Новый проект** (шаг 2 из 3)

Выбираем предметную область проекта

Наборы ролей:

☒ Стандартный

☒ Разработка Программного Обеспечения

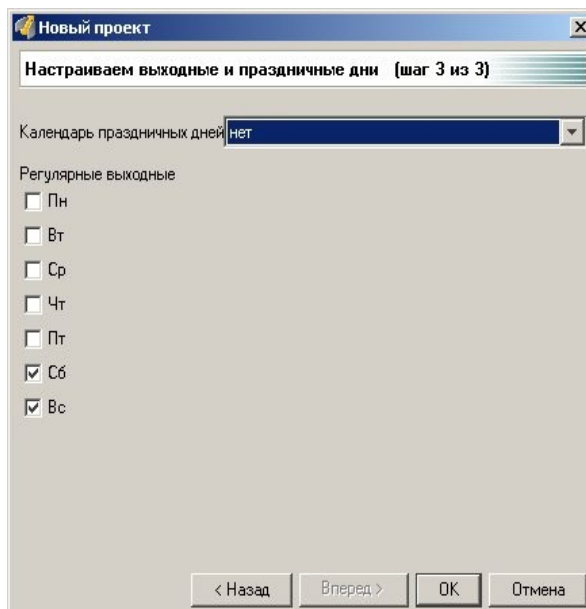
**SoftwareDevelopment**

- разработчик
- составитель документации
- тестер
- дизайнер по графике
- переводчик документации
- подготовка пакетов (.rpm, .tgz, ...)
- аналитик
- web дизайнер
- нет определенной роли

< Назад    Вперед >    ОК    Отмена

**Рисунок 4. Окно выбора предметной области проекта**

4. Следующий экран предполагает настройку выходных дней недели, а также праздников страны разработчика (рисунок 5). Поскольку России в перечне стран в календаре праздничных дней нет, оставляем настройки по умолчанию.



**Рисунок 5. Окно настройки выходных и праздничных дней**

5. Сразу сохраняем в указанном месте новый проект под заданным именем в файл, выбрав из главного меню пункт *Проект -> Сохранить*.

Выполненные установки никак не отражаются на внешнем виде окна диаграммы Ганта, поскольку влияют только на расчет параметров проекта.

### **3. Разработка ленточного и сетевого графика проекта**

Для построения сетевого графика проекта и задания исходных сведений о структуре этапов проекта часто используется диаграмма Ганта (отсюда и название изучаемого пакета - ***GanttProject***).

Диаграмма Ганта (англ. Gantt chart, также ленточная диаграмма, график Ганта) — это популярный тип столбчатых диаграмм, который используется для иллюстрации плана, графика работ по какому-либо проекту. Является одним из методов планирования проектов.

Первый формат диаграммы был разработан Генри Л. Гантом (Henry L. Gantt, 1861 - 1919) в 1910 году.

Диаграмма Ганта представляет собой отрезки (графические плашки), размещенные на горизонтальной шкале времени. Каждый отрезок соответствует отдельной задаче или подзадаче. Задачи и подзадачи, составляющие план, размещаются по вертикали. Начало, конец и длина отрезка на шкале времени соответствуют началу, концу и длительности задачи. На некоторых диаграммах Ганта также показывается зависимость между задачами. Диаграмма может использоваться для представления текущего состояния выполнения работ: часть прямоугольника, отвечающего задаче, заштриховывается, отмечая процент выполнения задачи; показывается вертикальная линия, отвечающая моменту «сегодня».

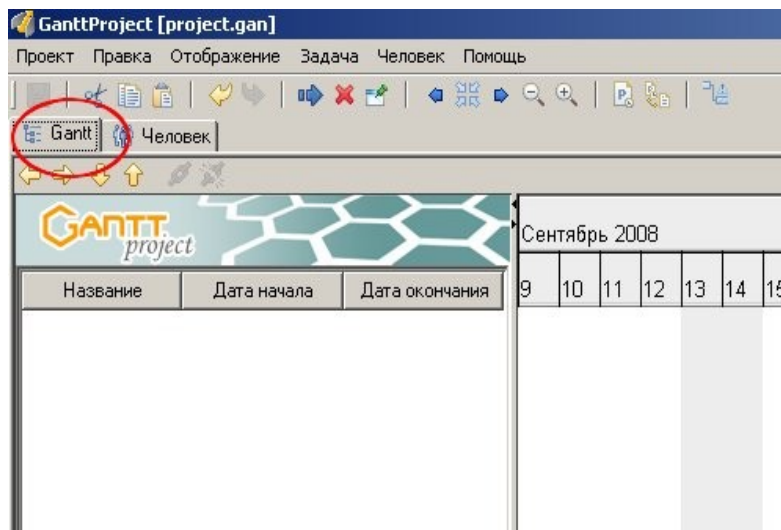
Часто диаграмма Ганта соседствует с таблицей со списком работ, строки которой соответствуют отдельно взятой задаче, отображенной на

диаграмме, а столбцы содержат дополнительную информацию о задаче.


Используя эскиз на рисунке 1, можно построить ленточный график проекта средствами GanttProject в окне ленточного графика на закладке *Gantt* (нужно помнить о том, что в GanttProject работам соответствуют вершины графика, а не дуги).

Для построения ленточного графика необходимо:

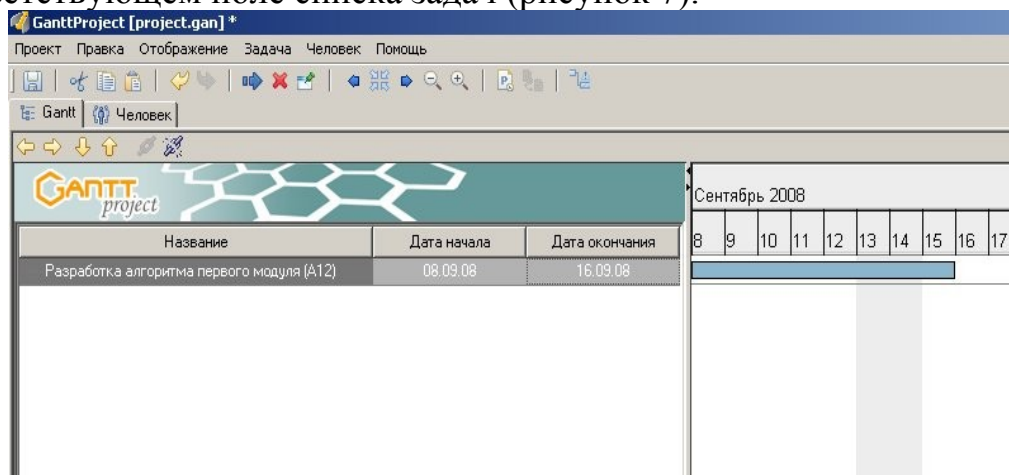
1. Перейти в окно ленточного графика; для этого перейти на закладку *Gantt* (рисунок 6)



**Рисунок 6. Переход в окно ленточного графика (закладка Gantt)**

2. В рабочем поле окна создать первую задачу. Сделать это можно разным способом: нажав на кнопку панели инструментов , выбрав пункт главного меню «Задача -> Новая задача», набрав на клавиатуре комбинацию «горячих клавиш» <Ctrl+T>, или щелкнув дважды левой кнопкой мыши по указателю в поле списка задач.

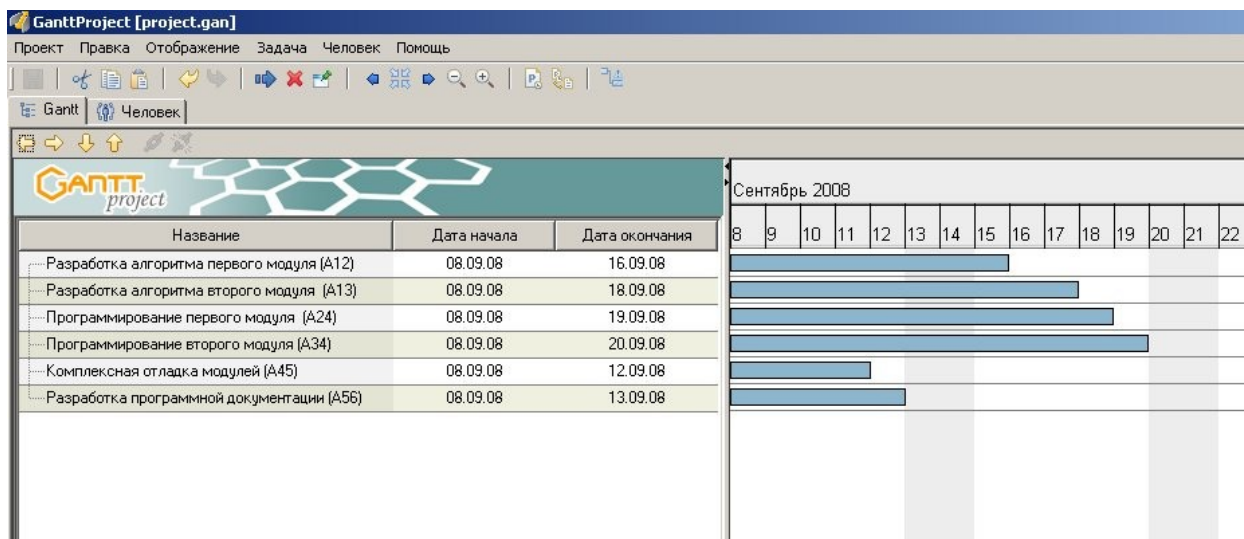
Для вновь созданной задачи необходимо указать её название, дату начала и дату окончания выполнения. Сделать это можно по двойному щелчку мыши в соответствующем поле списка задач (рисунок 7).



**Рисунок 7. Создание новой задачи**

3. Последовательно добавить остальные задачи проекта и ввести в них названия задач и их длительности.




В результате ленточный график должен выглядеть примерно так, как показано на рисунке 8.



**Рисунок 8. Перечень задач проекта на ленточном графике**

Необходимо заметить, что длительность задачи удобнее задавать в окне свойств задачи — количеством дней, а не непосредственным заданием диапазона «дата начала — дата окончания», так как диапазон должен учитывать праздничные и выходные дни недели, которые необходимо исключать при подсчете общей длительности выполнения задачи. Открыть окно свойств каждой задачи можно через пункт главного меню «Задача -> Свойства задачи» или через соответствующий пункт контекстного меню, предварительно указав требуемую задачу щелчком мыши.

4. Так как простое перечисление задач проекта не раскрывает его структуры, на ленточном графике требуется указать связи между задачами.

Для того чтобы связать две задачи на ленточном графике, достаточно щелкнуть мышкой на первой задаче и, удерживая нажатой левую кнопку мыши, перетянуть указатель на вторую задачу. Этот же результат можно получить и иначе: достаточно выбрать связываемые задачи и щелкнуть на панели инструментов под закладкой *Gantt* кнопку *Ссылка* . На той же панели инструментов располагается кнопка *Разъединить*  - для быстрого удаления связи между двумя предварительно выбранными задачами, а так же набор кнопок изменения уровня и положения задачи на диаграмме Ганта .

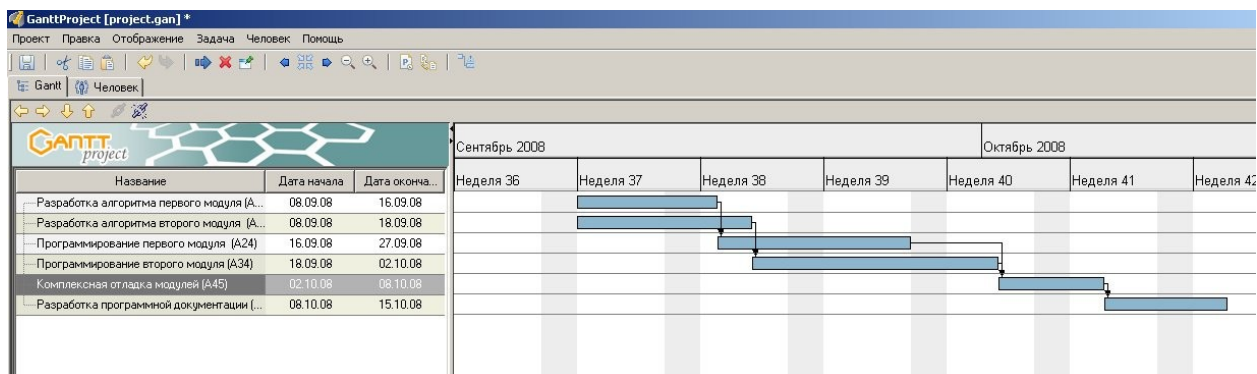
Задачи могут быть связаны друг с другом зависимостями четырех типов:

- Finish-Start (начать по завершению такой-то задачи).
- Finish-Finish (закончить не раньше такой-то задачи).
- Start-Finish (закончить не ранее начала такой-то задачи).
- Start-Start (начать не ранее начала такой-то задачи).

Изменить тип связи между задачами можно через диалог свойств задачи, в которую следует связь (рисунок 14).

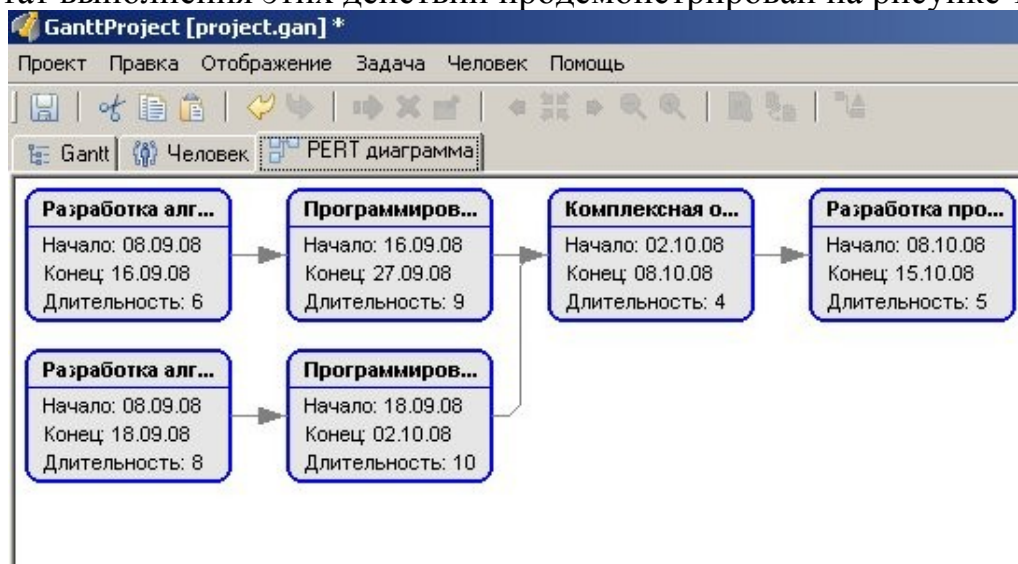
В результате распределение задач на временной оси будет автоматически скорректировано программой и первоначальный вариант ленточного графика будет сформирован (рисунок 9).





**Рисунок 9. Ленточный график проекта в среде GanttProject**

Ленточный график удобно отображает последовательность задач на временной оси, однако, если длительности выполнения задач достаточно велики начинает страдать наглядность структуры проекта. Для просмотра структуры проекта удобно использовать сетевой график. Для этого требуется активизировать закладку «*PERT диаграмма*», сделать это можно из главного меню программы, выбрав пункт *Отображение -> PERT диаграмма*. Результат выполнения этих действий продемонстрирован на рисунке 10.




**Рисунок 10. Сетевой график проекта**

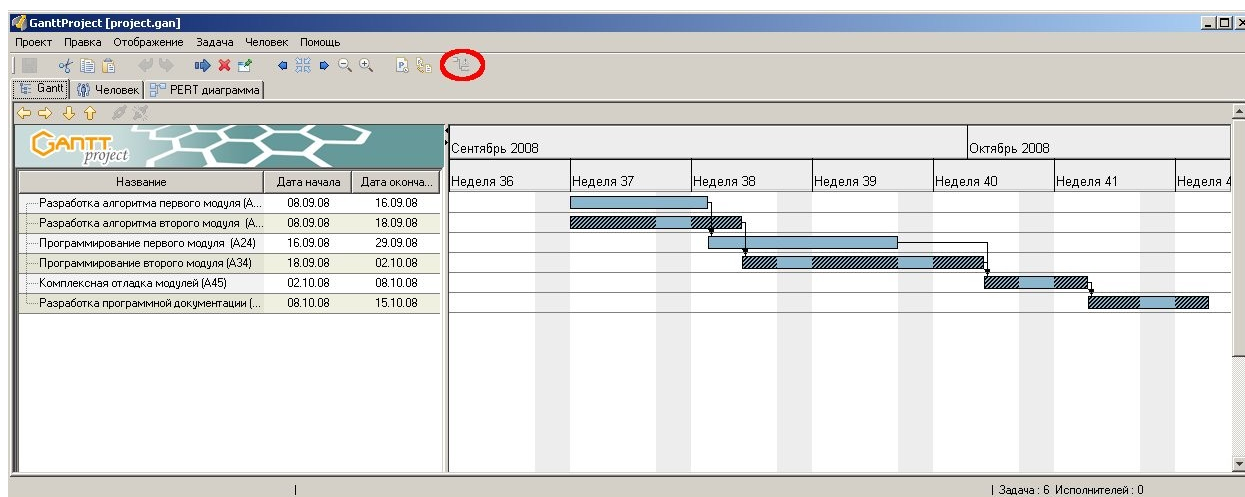
Построив сетевой график, сохраните файл проекта на диске, дав ему более или менее осмысленное имя, например, *Разработка ПО*. Название файла используется в качестве имени проекта и отображается в полосе заголовка окна сетевого графика.

#### 4. Уточнение взаимосвязи между задачами

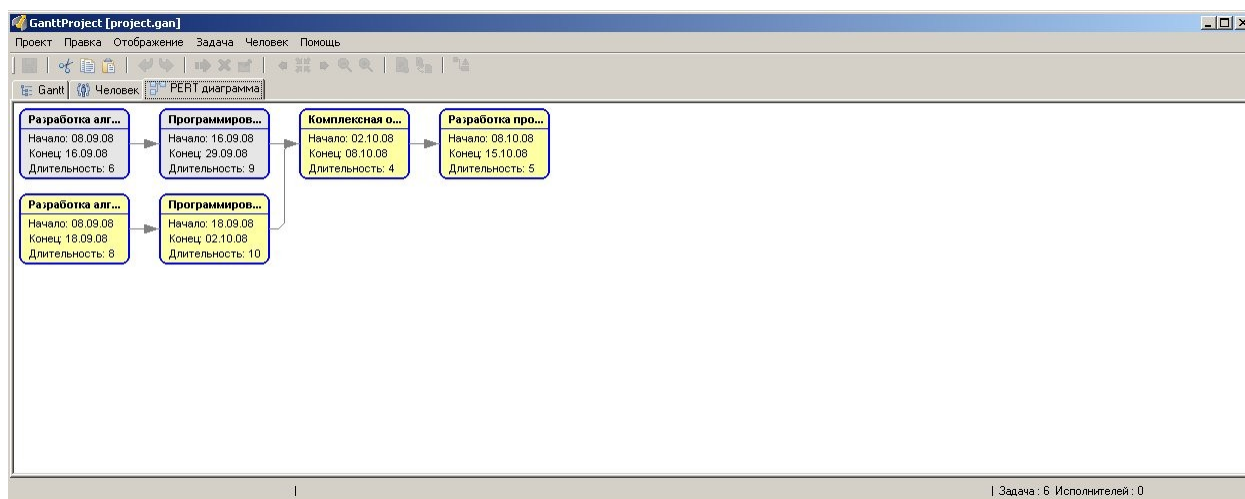
При построении сетевого графика проекта *GanttProject* может рассчитать и показать так называемый *критический путь*. Критический путь — это наиболее длительная последовательность задач от начала проекта до его окончания с учетом их взаимосвязи. Задачи лежащие на критическом пути (критические задачи) имеют нулевой резерв времени выполнения и в случае изменения их длительности изменяются сроки всего проекта. В связи с этим при выполнении проекта критические задачи требуют более тщательного контроля, в частности, своевременного выявления проблем и

рисков, влияющих на сроки их выполнения и, следовательно, на сроки выполнения проекта в целом. В процессе выполнения проекта критический путь проекта может меняться, так как при изменении длительности задач некоторые из них могут оказаться на критическом пути.

Для расчета критического пути выполнения проекта нажмите на кнопку  панели инструментов. Задачи, входящие в критический путь на ленточном графике (закладка *Gantt*) будут выделены штриховкой (рисунок 11), а на сетевом графике (закладка *PERT диаграмма*) изменят свой цвет (рисунок 12).



**Рисунок 11. Критический путь на ленточном графике проекта**



**Рисунок 12. Критический путь на сетевом графике проекта**

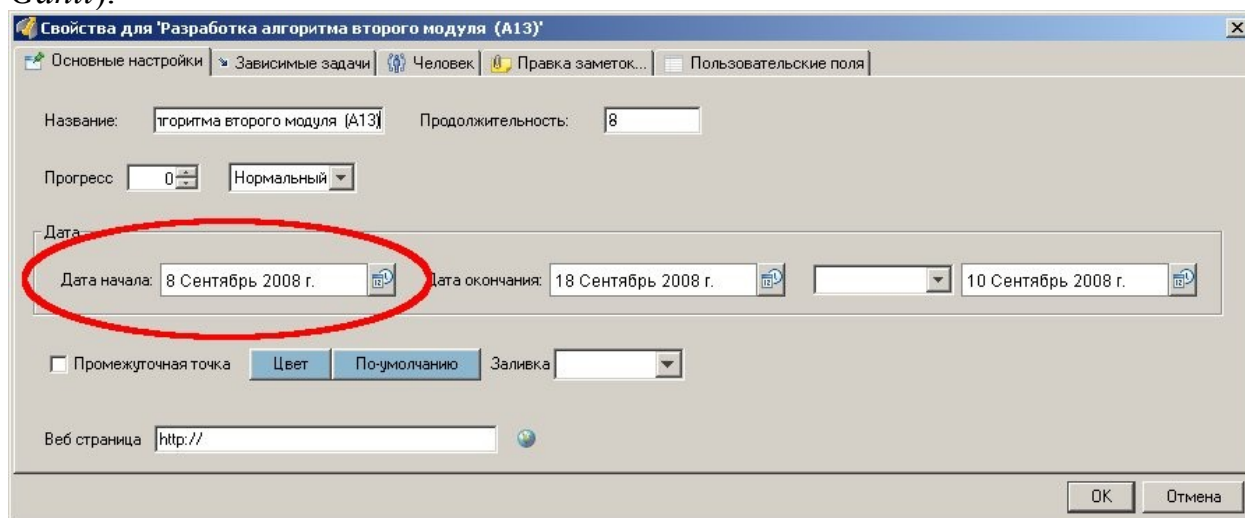
И хотя более тщательный анализ проекта возможен только после назначения имеющихся ресурсов, уже на этом этапе можно попытаться сократить длительность критического пути. Один из возможных способов — уточнение взаимосвязи между задачами.

Чтобы внести в проект соответствующие коррективы, необходимо предварительно проанализировать задачи, включенные в проект, и решить, какие из них можно совместить во времени (выполнять параллельно).

Начинать программировать модуль до тех пор, пока не завершена



разработка его алгоритма нельзя. Следовательно задачи A12 и A24 должны строго следовать друг за другом, как и задачи A13 и A34. А вот параллельно выполнять эти задачи парами вполне возможно, если у нас будет для этого достаточно ресурсов (исполнителей). В связи с этим, необходимо установить *дату начала* выполнения для задач A12 и A13 одной и той же. Сделать это можно в окне свойств задачи A13 (рисунок 13), вызвать которое можно дважды щелкнув по изображению задачи на ленточном графике (на закладке *Gantt*).



**Рисунок 13. Окно свойств задачи A13**

Далее можно установить, что перейти к совместной отладке модулей, пока не отлажен каждый из них в отдельности невозможно. Следовательно начать выполнять задачу A45 (комплексная отладка программы) можно только после полного завершения задач A24 и A34. Следовательно, совместить выполнение этих задач невозможно.

Вместе с тем, к оформлению программной документации можно приступить, например, после успешного завершения основных тестовых прогонов разработанной программы. Предположим, что на их проведение уйдет *первый день* из четырех, отведенных на комплексную отладку. Таким образом, две последние задачи проекта (A45 и A56) мы можем выполнять частично параллельно, с задержкой задачи-последователя (A56) на 1 день. Чтобы внести подобное изменение в проект, откроем свойства задачи A56, перейдём на закладку «Зависимые задачи», выберем из списка зависимых задач задачу A45, изменим тип связи с «Начать по завершению» на «Начать не раньше начала» (т.е. укажем возможность параллельного выполнения этих двух задач), добавим в поле «Задержка» единицу (задержка на один день) (рисунок 14).

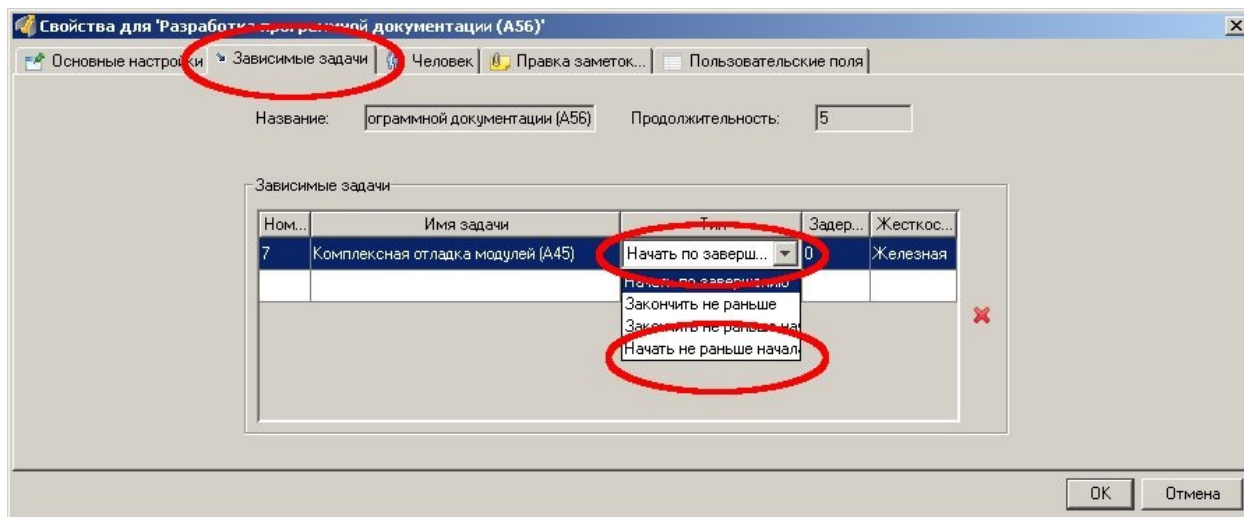


Рисунок 14. Окно свойств задачи A56

В результате выполненных действий ленточный график проекта (диаграмма Ганта) должен выглядеть как показано на рисунке 15. Не забудьте своевременно сохранить результаты выполнения проекта.

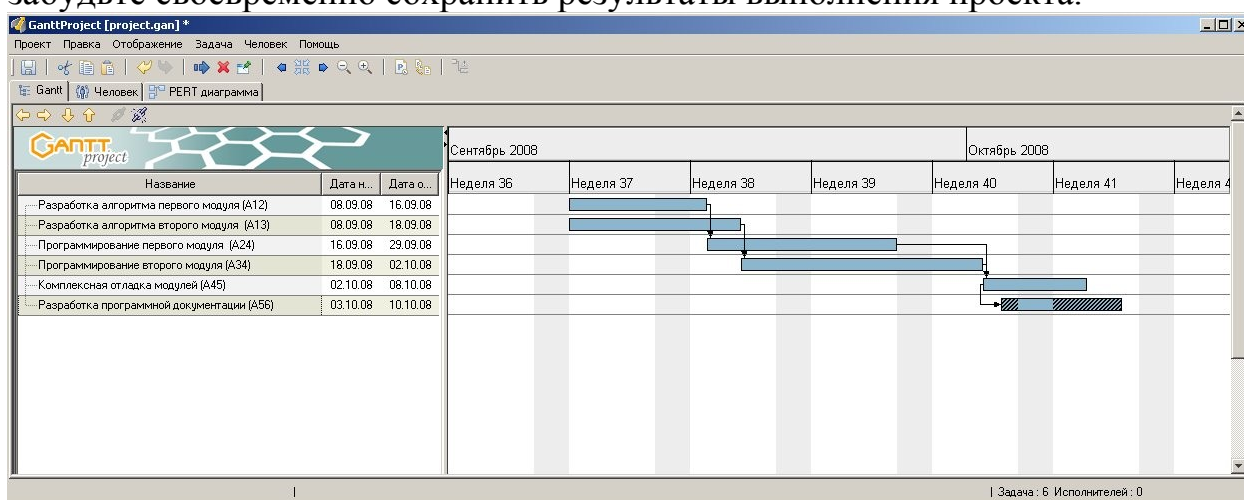
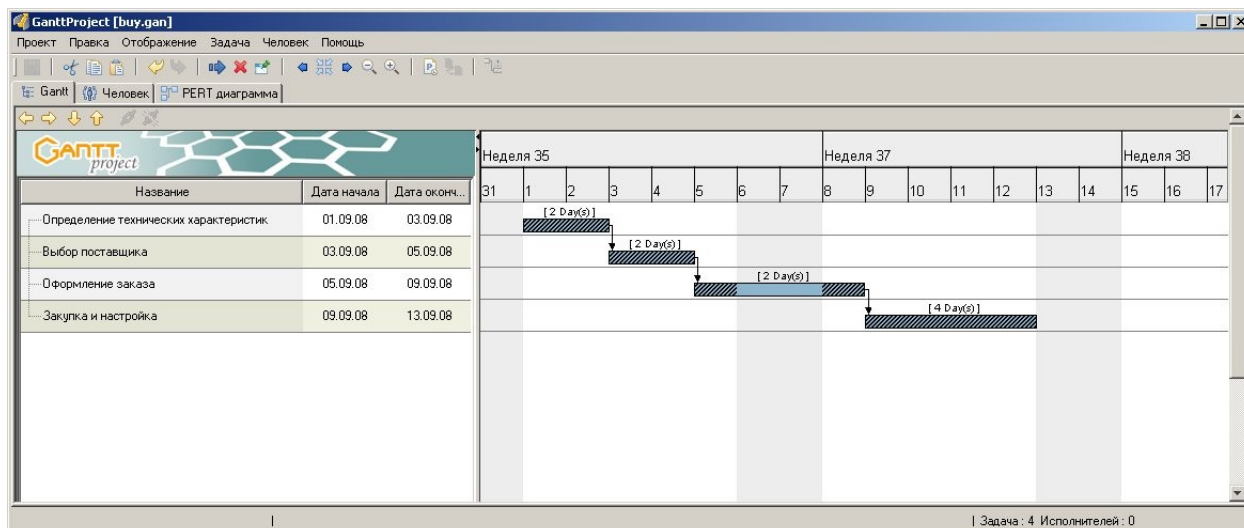


Рисунок 15. Диаграмма Ганта проекта

## 5. Создание взаимосвязанных проектов

Предположим, что разрабатываемый программный продукт требует для своего применения технического переоснащения организации. Тогда вполне вероятно, что мероприятия, связанные с поставкой и развертыванием новой техники, могут выполняться параллельно с разработкой программного продукта. Вследствие этого указанные мероприятия целесообразно выделить в отдельный проект. Параметры проекта *Закупка техники* и предполагаемый вид его календарного графика представлены на рисунке 16.

Самостоятельно создайте новый проект и заполните его диаграмму Ганта в соответствии с данными рисунка 16. Не забудьте сохранить результат в отдельном файле.



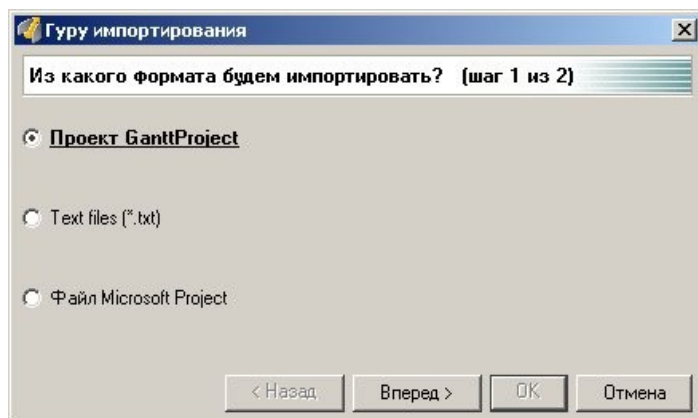
**Рисунок 16. Диаграмма Ганта проекта по закупке техники для разрабатываемого ПО**

В силу очевидной связанности двух рассматриваемых проектов, целесообразно объединить их в качестве составных частей одного главного проекта.

Выполним такое объединение на основе первого из созданных нами проектов - *Разработка ПО*.

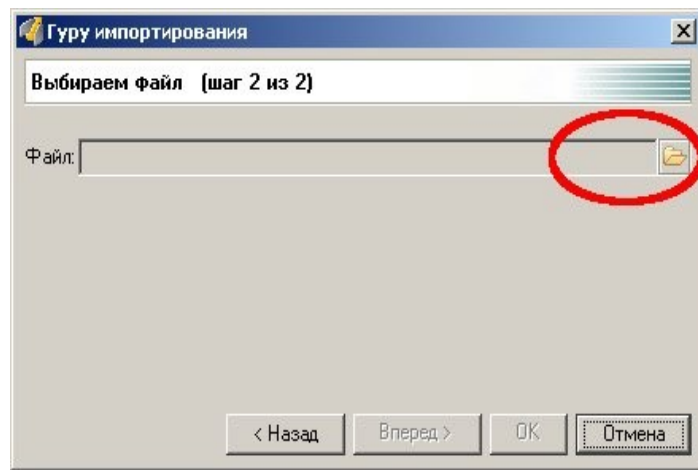
Для этого необходимо:

1. Открыть файл первого проекта *Разработка ПО*.
2. Выбрать пункт главного меню «Проект->Импортировать».
3. В появившемся окне диалога выбрать тип источника: Проект *GanttProject* (рисунок 17).



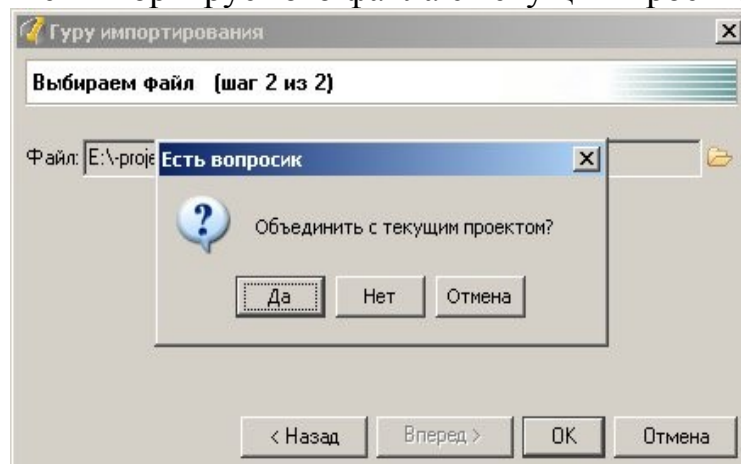
**Рисунок 17. Диалог импорта. Выбор типа проекта**

4. Нажав кнопку «Вперед >», на следующем шаге открыть диалог выбора файла проекта (рисунок 18) и выбрать файл со вторым созданным проектом *Закупка техники*.



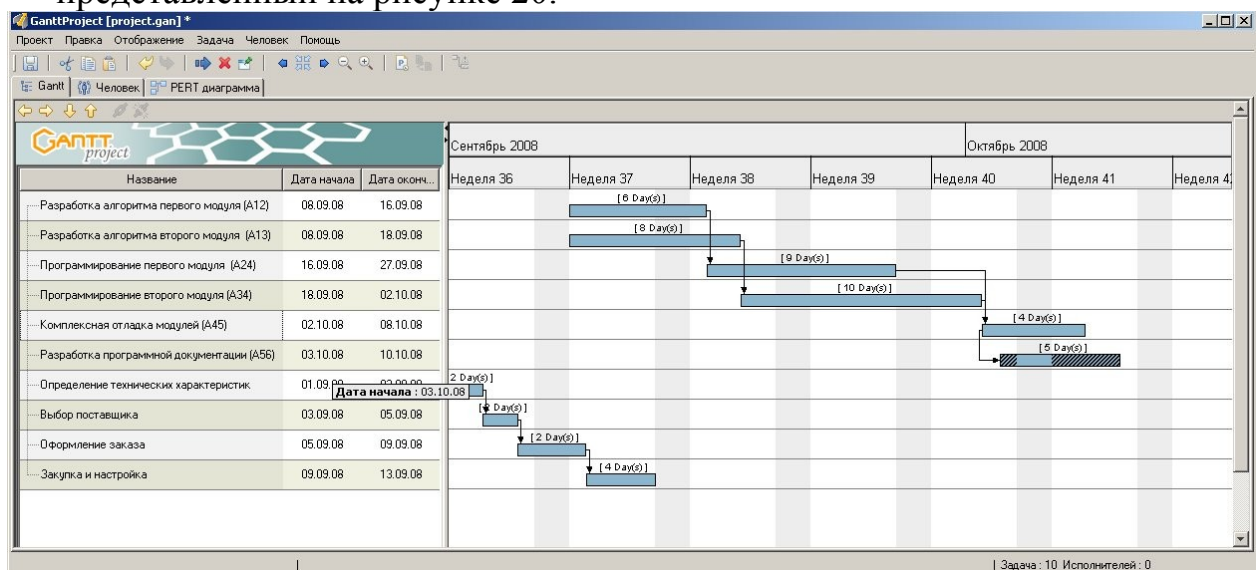
**Рисунок 18. Диалог импорта. Выбор файла проекта**

5. Нажав кнопку «Ок», перейти к заключительному шагу, подтвердив запрос на объединение импортируемого файла с текущим проектом (рисунок 19).



**Рисунок 19. Диалог импорта. Подтверждение объединения проектов**

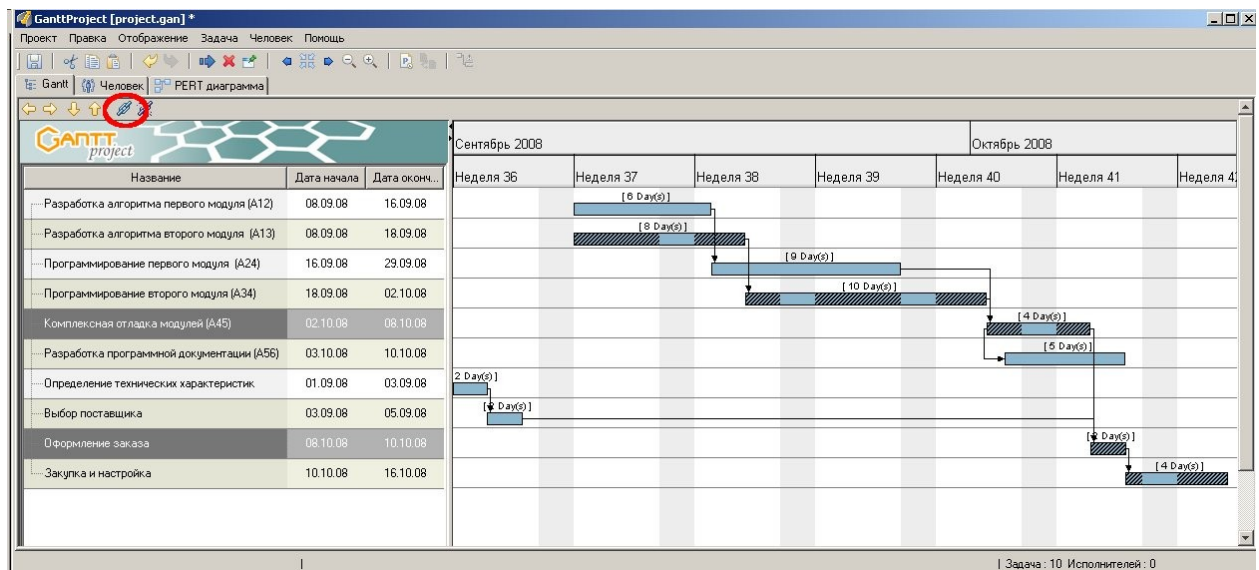
6. В результате объединения проектов должен получиться график проекта, представленный на рисунке 20.



**Рисунок 20. Диаграмма Ганта объединенного проекта**

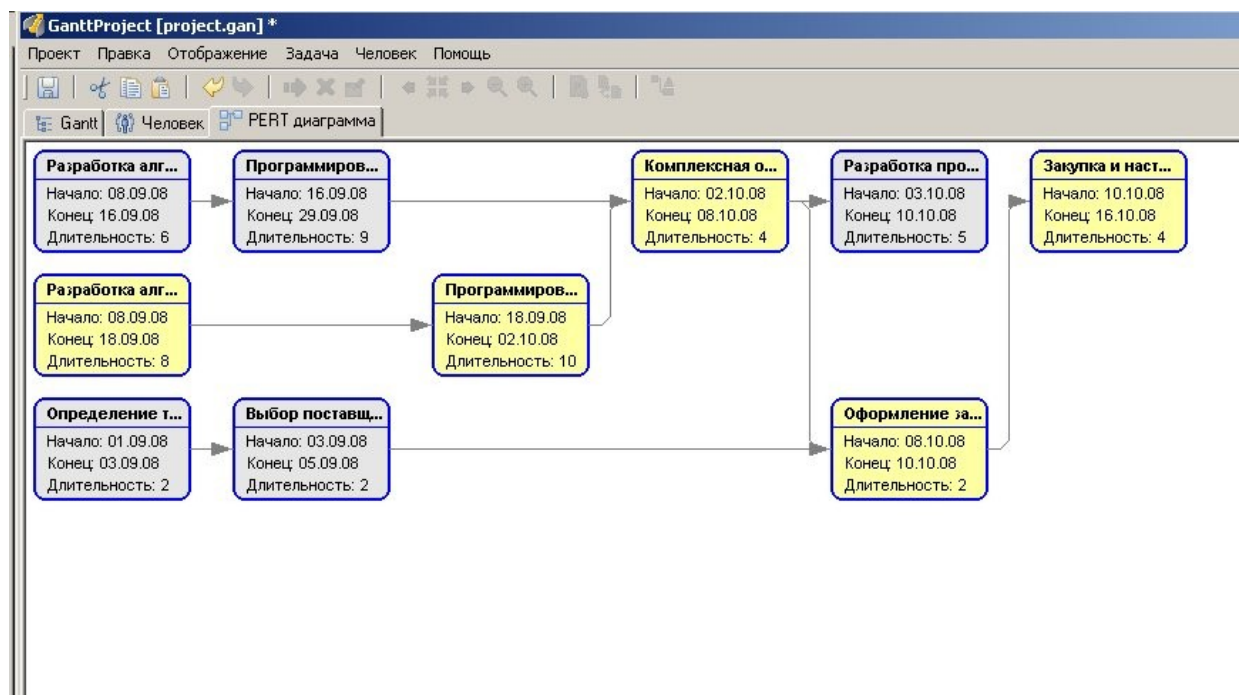
Предположим, что работа *Оформление заказа* проекта *Закупка техники* может быть начата только после завершения *комплексной отладки модулей* разрабатываемого ПО (работа *Комплексная отладка модулей*

проекта *Разработка ПО*). Чтобы отразить указанную зависимость в главном проекте, достаточно создать связь между соответствующими задачами проектов. Процедура связывания любых двух задач в данном случае аналогична связыванию задач в «обычном» проекте (см. выше). Результат связывания задач показан на рисунке 21.



**Рисунок 21. Результат связывания задач, относящихся к разным проектам**

Обратите внимание на критический путь объединенного проекта (рисунок 21, рисунок 22). Сохраните полученный результат на диске в файле с новым именем.



**Рисунок 22. Сетевой график объединенного проекта**