

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ им. И. РАЗЗАКОВА**

**Факультет информационных технологий**

**Кафедра «Программное обеспечение программных систем»**

**«Утверждаю»**

Проректор по учебной работе  
Сартов Т.Э. \_\_\_\_\_

«\_\_» \_\_\_\_\_ 20\_\_ г.

**«Согласовано»**

Ректор КГТУ им.И.Раззакова  
проф. Джаманбаев М.Дж.

«\_\_» \_\_\_\_\_ 20\_\_ г.

**ПРОГРАММА ГОСУДАРСТВЕННОГО ЭКЗАМЕНА ВЫПУСКНИКОВ  
ПО НАПРАВЛЕНИЮ  
710400 ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

**Профиль подготовки: Технология командной разработки программного обеспечения**

Разработана на основе ГОС по направлению 710400 Программная инженерия  
Программа рассмотрена и одобрена на заседании кафедры «Программное  
обеспечение компьютерных систем», протокол № от «\_\_» \_\_\_\_\_ 2017г.

Зав. кафедрой \_\_\_\_\_ профессор Тен И.Г

**Бишкек 2018г.**

## СОДЕРЖАНИЕ

1. Цель Государственного экзамена по специальности (направлению)
2. Общие требования к выпускнику, предусмотренные ГОС
3. Перечень практических задач, включенных в государственный экзамен и форма экзамена
4. Критерии оценки знаний студентов

### 1. Цель Государственного экзамена по направлению

Государственный итоговый экзамен является одной из форм аттестации выпускников и проводится в соответствии с Законом Кыргызской Республики «Об образовании», Государственным образовательным стандартом «Высшее образование. Общие требования», «Положением об итоговой аттестации выпускников высших учебных заведений Кыргызской Республики», Государственным образовательным по направлению 710400 Программная инженерия после завершения полного курса обучения согласно рабочего учебного плана и до начала выполнения выпускной квалификационной работы (дипломного проекта).

Государственный итоговый экзамен по направлению 710400 Программная инженерия - установить фактический уровень теоретической подготовки каждого выпускника к профессиональной деятельности, оценить умение применять знания для решения практических задач, ориентироваться в различных научно-технических ситуациях, связанных с профессиональной деятельностью и выполнением выпускной квалификационной работы.

### 2. Общие требования к выпускнику, предусмотренные ГОС

Выпускник по направлению подготовки **710400 - Программная инженерия** с присвоением академической степени "бакалавр" в соответствии с целями ООП и задачами профессиональной деятельности, указанными в пп. 3.4 и 3.8 настоящего ГОС ВПО, должен обладать следующими компетенциями:

#### - универсальными:

способен к приобретению новых знаний с большой степенью самостоятельности с использованием современных образовательных и информационных технологий; **(ОК-3)**

способен понимать и применять традиционные и инновационные идеи, находить подходы к их реализации и участвовать в работе над проектами, используя базовые методы исследовательской деятельности; **(ОК-4)**

#### -инструментальными (ИК):

способен к восприятию, обобщению и анализу информации, постановке цели и выборе путей ее достижения; **(ИК-1)**

способен логически верно, аргументировано и ясно строить свою устную и письменную речь на государственном и официальном языках; **(ИК-2)**

владеть одним из иностранных языков на уровне социального общения; **(ИК-3)**

способен осуществлять деловое общение: публичные выступления, переговоры, публичные выступления, проведение совещаний, деловую переписку, электронные коммуникации; **(ИК-4)**

владеет основными методами, способами и средствами получения, хранения и переработки информации, навыками работы с компьютером, как средством

управления информацией, в том числе в глобальных компьютерных сетях и корпоративных информационных системах; **(ИК-5)**

способен участвовать в разработке организационных решений. **(ИК-6)**

способен формализовать предметную область программного проекта и разработать спецификации для компонентов программного продукта; **(ПК-1)**

**-проектная деятельность:**

способен применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов; **(ПК-2)**

способен читать, понимать и выделять главную идею прочитанного исходного кода, документации; **(ПК-3)**

способен моделировать объектов; **(ПК-4)**

способен создавать программные интерфейсы **(ПК-5)**

**-технологическая деятельность:**

способен использовать операционные системы, сетевые технологии, средства разработки программного интерфейса, применять языки и методы формальных спецификаций, систем управления базами данных; **(ПК-6)**

**-производственная деятельность:**

способен применять основные методы и инструменты разработки программного обеспечения; **(ПК-7)**

способен понимать концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования); **(ПК-8)**

**-организационно-управленческая деятельность:**

способен понимать модели жизненного цикла, методы управления процессами разработки требований, проектирования, конструирования, тестирования, эволюции и сопровождения; **(ПК-9)**

**-Сервисно - эксплуатационная деятельность:**

способен понимать основные концепции и модели эволюции и сопровождения программного обеспечения, особенности эволюционной деятельности с технической точки зрения, реинженеринг и рефакторинг. **(ПК-10)**

### **3. Перечень дисциплин, включенных в государственный экзамен и форма экзамена**

Проверка умения и практических навыков программирования, моделирующая реальную ситуацию при найме выпускников на работу в software company, когда у соискателей-программистов оценивают умение и практические навыки разрабатывать программный проект за определенный срок. Эта часть экзаменационного билета должна включать практическое задание на разработку ПО для решения конкретной прикладной проблемы. Эти экзаменационные билеты со спецификациями задач составлены преподавателями кафедры, по выбранным кафедрой восьми учебным предметам, изученных бакалаврами в течении четырех лет обучения (МО, ТПР, Проектирование ПО, ФОР (структурное программирование/алгоритмический язык1), СУБД, ООП и Тестирование ПО). Должно быть заготовлено от 10-20 задач по каждому из выбранных восьми предметов.

Такая задача должна иметь трудоемкость по продолжительности времени разработки программы не более 80–120 минут. Проверка практических навыков программирования должна включать оценку умения и навыков бакалавров по стадиям разработки ПО, таких как:

- 1) Разработать спецификацию требований к ПО, в котором описать:
  - i. Бизнес-процесс и анализ расхождения;
  - ii. Бизнес-цель разработки программного обеспечения;
  - iii. Описать максимально широкий круг stakeholders ПО;
  - iv. Перечислить основные функции ПО;
- 2) Проектировать ПО (входной, выходной интерфейсы, блок-схемы алгоритмов, архитектуру ПО, тесты для проверки по всем ветвлениям логики программы);

- 3) Конструировать ПО (структура и взаимосвязь модулей программы, выбор и обоснование языка программирования, кодирование модулей ПО);
- 4) Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

Всего на гос экз по спецу предоставляется 85 билетов. В каждом билете описана спецификация одной задачи. Ниже приведены спецификации этих задач.

### Спецификация задачи №I.1

Найти корень произвольного нелинейного уравнения – ноль нелинейной функции  $f(x)$  с допустимой погрешностью  $Tolerance = 1E-28$  методом деления отрезка пополам (Bisection Method). Нелинейная функция  $f(x)$  имеет произвольный аналитический вид, составленный из математических функций (полиномов различных степеней, тригонометрических –  $\sin(x)$ ,  $\cos(x)$ ,  $\exp(x)$ ,  $\ln(x)$ ,  $\log(x)$  и. т. д.). Эта функция имеет математический смысл и существует хотя бы одно решение задачи. Описание метода деления отрезка пополам – Bisection Method – приведено ниже:

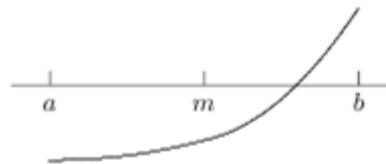
In finite-precision arithmetic, there may not be a floating-point number  $x$  such that  $f(x)$  is exactly zero. One alternative is to look for a very short interval  $[a, b]$  in which  $f$  has a change of sign, since the corresponding continuous function must be zero somewhere within such an interval. An interval for which the sign of  $f$  differs at its endpoints is called a *bracket*. The *bisection method* begins with an initial bracket and successively reduces its length until the solution has been isolated as accurately as desired. At each iteration, the function is evaluated at the midpoint of the current interval, and half of the interval can then be discarded, depending on the sign of the function at the midpoint. More formally, the algorithm is as follows, where  $\text{sign}(x) = 1$  if  $x \geq 0$  and  $\text{sign}(x) = -1$  if  $x < 0$ :

Initial input: a function  $f$ , an interval  $[a, b]$  such that  $\text{sign}(f(a)) \neq \text{sign}(f(b))$ , and an error tolerance  $tol$ .

```

while ((b - a) > tol) do
  m = a + (b - a)/2
  if sign(f(a)) = sign(f(m)) then
    a = m
  else
    b = m
  end
end
end

```



#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### Спецификация задачи №I.2

Найти корень произвольного нелинейного уравнения – ноль нелинейной функции  $f(x)$  с допустимой погрешностью  $Tolerance = 1E-28$  методом Ньютона (Newton's Method). Нелинейная функция  $f(x)$  имеет произвольный аналитический вид, составленный из математических функций (полиномов различных степеней, тригонометрических –  $\sin(x)$ ,  $\cos(x)$ ,  $\exp(x)$ ,  $\ln(x)$ ,  $\log(x)$  и т. д.). Эта функция имеет математический смысл и существует хотя бы одно решение задачи. Описание метода Ньютона – Newton's method – приведено ниже:

Mathematical description of the Newton's algorithm
$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)} \text{ for all } k = 0, 1, \dots;$
$x^0$ – Initial approximation; $\varepsilon$ – Tolerance; $f(x^k)$ – the objective function at point $x^k$ ; $f'(x^k)$ – the first derivative of the objective function at point $x^k$ .

### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес-процесса).

### Спецификация задачи №1.3

Найти значение переменной  $x^*$ , которое минимизирует (или максимизирует) произвольную целевую функцию  $f(x)$  с допустимой погрешностью  $Tolerance = 1E-28$  методом Поразрядного поиска. Целевая функция  $f(x)$  имеет произвольный аналитический вид, составленный из математических функций (полиномов различных степеней, тригонометрических –  $\sin(x)$ ,  $\cos(x)$ ,  $\exp(x)$ ,  $\ln(x)$ ,  $\log(x)$  и т. д.). Эта функция имеет математический смысл и существует хотя бы одно решение задачи. Описание метода Поразрядного поиска – Pocket Search method – приведено ниже:

Mathematical description of the Pocket Search algorithm
<pre>  IF <math>f^k \geq f^{k-1}</math> /*Pocket Search Method to find a <i>minimum</i> */    THEN      IF <math> h^k  \leq \varepsilon/R</math>        THEN <math>h^{k+1} = h^k; x^{k+1} = x^k; f^{k+1} = f^k;</math>        ELSE <math>h^{k+1} = -h^k/R; x^{k+1} = x^k + h^{k+1}; f^{k+1} = f(x^{k+1});</math>      endIF    ELSE <math>h^{k+1} = h^k; x^{k+1} = x^k + h^{k+1}; f^{k+1} = f(x^{k+1});</math>  endIF </pre>
for $k=1, 2, \dots;$
$x^0$ – Initial approximation of the solution; $h^0$ – Initial Increment (Initial Step-size of search); $\varepsilon$ – Tolerance; $R$ – Adjustable parameter of the method (Significance of Digit Position); $f^k = f(x^k)$ – objective function at point $x^k$ .

### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

#### Спецификация задачи №I.4

Найти значение переменной  $x^*$ , которое минимизирует (или максимизирует) произвольную целевую функцию  $f(x)$  с допустимой погрешностью  $Tolerance = 1E-28$  методом Ньютона. Целевая функция  $f(x)$  имеет произвольный аналитический вид, составленный из математических функций (полиномов различных степеней, тригонометрических –  $\sin(x)$ ,  $\cos(x)$ ,  $\exp(x)$ ,  $\ln(x)$ ,  $\log(x)$  и т. д.). Эта функция имеет математический смысл и существует хотя бы одно решение задачи. Описание метода Ньютона – Newton's method – приведено ниже:

Mathematical description of the Newton's algorithm
$x^{k+1} = x^k - \frac{f'(x^k)}{f''(x^k)} \text{ for all } k = 0, 1, \dots;$
$x^0$ – Initial approximation; $\varepsilon$ – Tolerance; $f(x^k)$ – the objective function at point $x^k$ ; $f'(x^k)$ – the first derivative of the objective function at point $x^k$ ; $f''(x^k)$ – the second derivative of the objective function at point $x^k$ .

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

#### Спецификация задачи №I.5

Найти значение переменной  $x^*$ , которое минимизирует (или максимизирует) произвольную целевую функцию  $f(x)$  с допустимой погрешностью  $Tolerance = 1E-28$  методом Золотого сечения. Целевая функция  $f(x)$  имеет произвольный аналитический вид, составленный из математических функций (полиномов различных степеней, тригонометрических –  $\sin(x)$ ,  $\cos(x)$ ,  $\exp(x)$ ,  $\ln(x)$ ,  $\log(x)$  и т. д.). Эта функция имеет математический смысл и существует хотя бы одно решение задачи. Описание метода Золотого сечения – Golden Section Search method – приведено ниже:

Mathematical description of the Golden Section Search algorithm
The function $f(x)$ is given in an interval $[a, b]$ on which $f(x)$ is unimodal; $r := (\sqrt{5} - 1)/2$ ; $x_1 := a + (1 - r)(b - a)$ ;

```

f1:= f(x1);
x2:= a + r(b-a);
f2:= f(x2);
WHILE ((b-a) > tol DO
|   |IF (f1 > f2) THEN /*Find a minimum */
|   |   a:= x1;
|   |   x1:= x2;
|   |   f1:= f2;
|   |   x2:= a + r(b-a);
|   |   f2:= f(x2);
|   |   ELSE
|   |   b:= x2;
|   |   x2:= x1;
|   |   f2:= f1;
|   |   x1:= a + (1-r)(b-a);
|   |   f1:= f(x1);
|   |ENDIF
|ENDWHILE

```

[a, b] – an interval on which f(x) is unimodal;  
tol – Tolerance;  
*f(x<sup>k</sup>)* – the objective function at point x<sup>k</sup>.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

**Спецификация задачи №I.6**

Фермеру необходимо выбрать какой из трех сельхозкультур люцерну, ячмень или пшеницу он должен выращивать на своем одно-гектарном поле. Для удобства будем обозначать эти сельхозкультуры через символы А, В, и С соответственно. Прибыль от каждой сельхозкультуры зависит от осадков в течение вегетационного периода. Фермер распределил категориям количество осадков по трем: как существенные, средние или слабые. Он оценивает свою будущую прибыль от каждой сельхозкультуры в зависимости от количества осадков, как это показано в Таблице 3.1 Основываясь на предыдущих наблюдениях за погодой и текущих прогнозах на предстоящий сезон фермер предполагает, что имеются одинаковые шансы на то, что осадки будут либо слабые либо более сильные (то есть средние или существенные). Кроме того он верит, что если слабых осадков не случится, то имеется шанс два к трем, что осадки будут скорее существенные чем средние. Какую сельхозкультуру фермер должен выращивать, чтобы максимизировать свою среднюю годовую ожидаемую прибыль?

Таблица 3.1 Условные прибыли

Средние прибыли в,\$			
Осадки/Rainfall	Сельхозкультура А	Сельхозкультура В	Сельхозкультура С
Существенные	7000	2500	4000
Средние	3500	3500	4000

Слабые	1000	4000	3000
--------	------	------	------

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

**Спецификация задачи №I.7**

Строительный кран специального назначения имеет деталь, которая подвержена случайной поломке. Эта деталь может быть закуплена про запас по цене 1000 долларов каждая в момент покупки самого крана у производителя, и остаточная стоимость такой запчасти составляет 200 долларов. Ежегодные затраты на содержание этой запчасти на складе включают: –затраты на хранение, –затраты на обработку детали на складе, –затраты, связанные с замораживанием финансовых средств в закупленных запчастях, и другие оцениваются в 25% от цены на саму запчасть. Если запчасть ломается и не будет на складе запаса этой запчасти, то ее производство по специальному заказу будет стоить 2000 долларов за штуку, а штрафы за простой крана оцениваются в 10 000 долларов. Поломки этой запчасти происходят полностью случайным образом со средним временем между двумя поломками 2.5 года. Кран нужен в работе в течение двухлетнего периода. Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь с допустимой погрешностью 0.01 при определении валидности свойства событий, образующих полную группу.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

**Спецификация задачи №I.8**

Предположим, что вы хотите вложить на фондовой бирже 10 000 долларов в акции одной из двух компаний: А или В. Акции компании А являются рискованными, но могут принести 50% прибыли от суммы инвестиции на протяжении следующего года. Если условия фондовой биржи будут неблагоприятны, сумма инвестиции может обесцениться

на 20%. Компания В обеспечивает безопасность инвестиции с 15% прибыли в условиях повышения котировок на бирже и только 5% – в условиях понижения котировок. Все аналитические публикации, с которыми могут ознакомиться (а они всегда есть в изобилии в конце года), с вероятностью 60% прогнозируют повышение котировок и с вероятностью 40% – понижение котировок. В какую компанию следует вложить деньги?

Информация, связанная с принятием решения, представлена в следующей таблице.

Возможные решения: Вложить деньги в:	Прибыли от инвестиции за один год	
	При повышении котировок (\$)	При понижении котировок (\$)
Акции компании А	5000	-2000
Акции компании В	1500	500
Вероятность события	0.6	0.4

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес-процесса).

### Спецификация задачи №1.9

Фирма планирует производство новой продукции быстрого питания в национальном масштабе. Исследовательский отдел убежден в большом успехе новой продукции и хочет внедрить ее немедленно, без рекламной кампании на рынках сбыта фирмы. Отдел маркетинга положение вещей оценивает иначе и предлагает провести интенсивную рекламную кампанию. Такая кампания обойдется в 100 000 долларов, а в случае успеха принесет 950 000 долларов годового дохода. В случае неуспеха рекламной кампании (вероятность этого составляет 30%) годовой доход оценивается лишь в 300 000 долларов. Если рекламная кампания не проводится вовсе, годовой доход оценивается в 400 000 долларов при условии, что покупателям понравится новая продукция (вероятность этого равна 0.8) и в 200 000 долларов с вероятностью 0.2, если покупатели останутся равнодушными к новой продукции.

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет

не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### Спецификация задачи №I.10

Фирма Intel производит партии продукции с определенным процентом 0.8%, 1%, 1.2% и 1,4% бракованных изделий с вероятностями 0.4, 0.3, 0.25 и 0.05 соответственно. Три потребителя продукции фирмы Intel Компания-А, Компания-В и Компания-С заключили с фирмой Intel контракт на получение партий изделий с процентом некачественных изделий не выше 0.8%, 1.2% и 1.4% соответственно. Фирма Intel штрафует компании: А, В и С в сумме 1000 долларов за каждый пункт процента (пункт процента - это одна десятая процента) в случае, когда процент некачественных изделий выше указанного. Наоборот, поставка партий изделий с меньшим процентом бракованных изделий, чем оговорено в контракте с компаниями: А, В и С, приносит фирме Intel соответственно премию в 2000, 1000 и 500 долларов за каждый пункт процента. Предполагается, что партии изделий перед отправкой не проверяются. Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### Спецификация задачи №I.11

Ежедневный спрос на булочки в продовольственном магазине задается следующим распределением вероятностей:

Величина спроса	100	150	200	250	300
Вероятность	0,1	0,35	0,3	0,15	0,1

Магазин покупает булочку по 55 центов, а продает по 1.20 доллара. Если булочка не будет продана в тот же день, то к концу дня она может быть реализована за 25 центов. Величина заказа булочек может принимать одно из возможных значений спроса, которые перечислены выше.

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет

не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### **Спецификация задачи №I.12**

Допустим, вы являетесь автором романа, который обещает быть популярным. У вас имеется возможность либо самостоятельно напечатать роман, либо через издательство. Издательство предлагает вам 20 000 долларов за подписание контракта. Если роман будет пользоваться спросом, будет продано 200 000 экземпляров, в противном случае - лишь 10 000 экземпляров. Издательство выплачивает авторский гонорар в сумме один доллар за экземпляр. Исследование рынка, проведенное издательством, свидетельствует о том, что существуют 70%-ная вероятность, что роман будет популярным. Если же вы сами напечатаете роман, то понесете потери в сумме 90 000 долларов, связанные с печатанием и маркетингом, но в этом случае каждый проданный экземпляр принесет вам прибыль в два доллара.

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### **Спецификация задачи №I.13**

Симметричная монета подбрасывается три раза. Вы получаете один доллар за каждое выпадение герба (Г) и дополнительно 0.25 доллара за каждые два последовательных выпадения герба (заметим, что выпадение ГГГ состоит из двух последовательностей ГГ). Однако Вам приходится платить 1.1 доллара за каждое выпадение решетки (Р). Вашим решением является участие или неучастие в игре.

а) Постройте соответствующее дерево решений для описания игры.

б) Будете ли вы играть в эту игру?

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет

не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### Спецификация задачи №I.14

Предположим, у вас имеется возможность сыграть в игру следующего содержания. Симметричная игральная кость бросается два раза, при этом возможны четыре исхода:

- выпадает два четных числа,
- выпадает два нечетных числа
- сначала четное число показывается, затем нечетное число показывается
- сначала нечетное число показывается, а затем четное число.

Вы можете делать одну ставку на два исхода. Например, вы можете поставить на два четных числа (исход 1) и два нечетных (исход 2). Выигрыш на каждый доллар, поставленный на первый исход, равен 2.2 доллара, на второй и третий исходы - 1.95 доллара, на четвертый - 1.50

- а) Постройте дерево решений для описанной игры
- б) На какие исходы следует делать ставки?

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### Спецификация задачи №I.15

Фирма планирует открыть новое предприятие в Арканзасе. В настоящее время имеется возможность построить либо крупное предприятие, либо небольшое, которое через два года можно будет расширить при условии высокого спроса на выпускаемую продукцию. Рассматривается задача принятия решений на десятилетний период. Фирма оценивает, что на протяжении этих 10 лет, вероятность высокого и низкого спроса на производимую продукцию будет равна 0.75 и 0.25 соответственно. Стоимость немедленного строительства крупного предприятия равна 5 миллионов долларов, а небольшого - один миллион долларов. Расширение малого предприятия через два года обойдется фирме в 4.2 миллиона долларов. Доход, получаемая от функционирования производственных мощностей на протяжении 10 лет, приводится в следующей таблице.

Альтернатива	Ожидаемый доход в течение года (тысяч долларов)	
	Высокий спрос	Низкий спрос
Большое предприятие сейчас	1000	300
Небольшое предприятие сейчас	250	200
Расширенное предприятие через 2 года	1000	300

- a) Постройте соответствующее дерево решений, принимая во внимание то, что через два года фирма может или расширить небольшое предприятие, или не расширять его.
- b) Сформулируйте стратегию строительства для фирмы в течение запланированного 10-летнего периода. (Для простоты не учитывать возможную информацию, появившуюся в течение этого периода).

Что требуется сделать?

Найти решение задачи по критерию Средней ожидаемой прибыли (Математическое ожидание прибыли) и по критерию Средних ожидаемых потерь.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders, то есть показать, что программа проверяет не только валидность входных данных, но и показать, что логика программы основана на валидных моделях реального бизнес–процесса).

### **Спецификация задачи №II. 2**

Вычислить значение числа в последовательности Фибоначчи по заданному номеру в ряду. Нумерация начинается с нуля.

**Числа Фибоначчи**— это элементы последовательности:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, ...,

в которой первые два числа равны 0 и 1, а каждое последующее число равно сумме двух предыдущих чисел.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграмму деятельности);
4. Рассчитать цикломатическую сложность программы;
5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация программы (показать, что программа удовлетворяет требованиям stakeholders).

### **Спецификация задачи №II.3**

Найти элемент в любой последовательности (чисел или символов), используя алгоритм бинарного поиска. Для сортировки массива можно использовать готовую функцию.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности);
4. Рассчитать цикломатическую сложность программы;
5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация программы (показать, что программа удовлетворяет требованиям stakeholders).

#### Спецификация задачи №II.4

Создать программу-калькулятор, реализующую следующие функции:

- Сложение
- Вычитание
- Умножение
- Деление
- Возведение в степень
- Расчет натурального логарифма
- Определение остатка от числа
- Перевод в двоичный код

Форма программы должна содержать текстовые поля для операндов и результата. Валидация входных данных обязательна.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое реализует вычисление. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграмму деятельности);
4. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Провести автоматический анализ кода
7. Валидация программы (показать, что программа удовлетворяет требованиям stakeholders).

#### Спецификация задачи №II.5

Создать программу, обучающую школьников решению квадратного уравнения. Программа должна предусматривать ввод школьником не только коэффициентов уравнения и вывод правильного решения, но и проверку его собственных результатов.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности, классов, последовательности);

4. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Провести автоматический анализ кода
7. Валидация программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи №II.6

Создать программу для работы с матрицами.

Операции: произведение, сложение и транспонирование. Пользователь может вводить матрицу любой размерности. Необходимо предусмотреть валидацию входных данных.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое реализует вычисление. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности);
4. Рассчитать метрики кода;
5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи №II.7

Разработать приложение, реализующее вычисление интеграла методом прямоугольников. Подинтегральная функция, границы интегрирования и количество разбиений интервала интегрирования задаются пользователем.

#### Метод прямоугольников:

Функция  $y=f(x)$  интегрируема на сегменте  $[a,b]$  и требуется вычислить ее интеграл

$$\int_a^b f(x) dx$$

Этот отрезок делится точками  $x_0, x_1, \dots, x_{n-1}, x_n$  на  $n$  равных отрезков длиной

$$\Delta x = \frac{b - a}{n}.$$

Формула средних прямоугольников имеет вид:

$$\int_a^b f(x) dx \approx h \sum_{i=1}^n f\left(x_{i-1} + \frac{h}{2}\right) = h \sum_{i=1}^n f\left(x_i - \frac{h}{2}\right),$$

где

$$h = \frac{b - a}{n}$$

**Требуется:**

1. Разработать спецификацию требований к ПО, которое реализует вычисление.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Рассчитать метрики кода;
5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №II.8

Разработать приложение, реализующее вычисление интеграла: методом Симпсона. Подинтегральная функция, границы интегрирования и количество разбиений интервала интегрирования задаются пользователем.

**Метод Симпсона:**

Функция  $y=f(x)$  интегрируема на сегменте  $[a,b]$  и требуется вычислить ее интеграл

$$\int_a^b f(x) dx$$

Метод Симпсона имеет вид:

$$\int_a^b f(x) dx \approx \frac{b-a}{6n} \cdot [(y_0 + y_{2n}) + 2(y_2 + y_4 + \dots + y_{2n-2}) + 4(y_1 + y_3 + \dots + y_{2n-1})]$$

**Требуется:**

1. Разработать спецификацию требований к ПО, которое реализует вычисление.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Рассчитать метрики кода;

5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №П.9

Разработать приложение, реализующее решение нелинейного уравнения вида:

$$x=f(x)$$

методом простых итераций с точностью  $\epsilon=10^{-3}$ . Функция  $f(x)$  и начальное приближение задается пользователем.

#### Метод простых итераций:

Пусть известно начальное приближение корня  $x=c_0$ . Подставляя это значение в правую часть уравнения  $x=f(x)$ , получаем новое приближение:  $c_1=f(c_0)$ . Далее, подставляя каждый раз значение корня, получаем новое приближение согласно формуле:

$$c_n=f(c_{n-1}).$$

Итерационный процесс прекращается, когда выполняется условие:

$$|c_{n+1} - c_n| < \epsilon.$$

#### Требуется:

1. Разработать спецификацию требований к ПО, которое реализует вычисление. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности);
4. Рассчитать цикломатическую сложность программы;
5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №П.10

Разработать приложение, реализующее вычисление интеграла методом трапеций. Подинтегральная функция, границы интегрирования и количество разбиений интервала интегрирования задаются пользователем.

#### Метод трапеций:

В общем случае формула трапеций принимает вид

$$\int_a^b f(x)dx \approx h \left( \frac{f_0 + f_n}{2} + \sum_{i=2}^{n-1} f_i \right) = \frac{b-a}{n} \left( \frac{f_0 + f_n}{2} + \sum_{i=2}^{n-1} f_i \right),$$

**Требуется:**

1. Разработать спецификацию требований к ПО, которое реализует вычисление.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграмму деятельности);
4. Рассчитать цикломатическую сложность программы;
5. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
6. Провести тестирование интерфейса, системное тестирование, профилирование программы;
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders)

**Спецификация задачи №II.11**

Создать программу - переводчик для двух языков для детей. Программа должна предусматривать выбор направления перевода (например, с русск на англ). Исходные слова должны выбираться из выпадающего списка. Можно использовать рисунки и текст. Результат должен приводиться в текстовом поле или в виде рисунка. В окне переводчика должны быть предусмотрены кнопки для быстрого вызова команд.

**Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Провести автоматический анализ кода
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

**Спецификация задачи №II.12**

Создать систему тестирования знаний студентов. Администратор вводит предметы и преподавателей. Преподаватель вводит вопросы и варианты правильных ответов. Для каждого вопроса 4 варианта ответов и только один правильный. Программа подсчитывает баллы и выдает результат по 100 балльной системе.

**Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №II.13

Создать систему для куратора учебной группы в целях контроля сдачи текущих экзаменов студентов. По введенным в систему результатам текущей сессии куратор должен увидеть, какие у каждого студента задолжности и какой процент успеваемости по группе за текущую сессию.

#### Требуется:

1. Разработать спецификацию требований к ПО. Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Провести автоматический анализ кода
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №II.14

Создайте программу – калькулятор для официанта кафе «Мороженное». Вводятся виды мороженого и их цены. Вводится текущий заказ (вид мороженого и количество порций). Программа рассчитывает текущий заказ, добавляет 5% от суммы заказа за обслуживание, рассчитывает общую сумму к оплате клиента, по оплаченному заказу подсчитывает сдачу клиенту, на конец дня выдает общую сумму выручки.

#### Требуется:

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №II.15

Создать систему по учету поездок сотрудников предприятия в разные города, а также расходов, связанных с этими командировками. В любой момент руководитель предприятия должен иметь возможность посмотреть информацию, связанную с командировкой сотрудника (Ф.И.О. должность, отдел, в котором он работает), данные об его командировке (организация, куда он командировается, количество дней командировки, цена билета туда и обратно, суточные) и общие расходы, связанные с ней ( $\text{Сумма} = \text{Дни} * \text{Суточные} + \text{Цена билетов}$ ). Кроме того, руководитель должен иметь возможность посмотреть отчет по расходам на командировки всего предприятия за заданный период.

#### Требуется:

1. Разработать спецификацию требований к ПО.

- Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты.
  3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
  4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
  5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
  6. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

#### **Спецификация задачи №П.16**

Создать систему для продовольственного отдела магазина, которая позволит вести учет продаж в отделе. В любой момент продавец должен знать, сколько у него в наличии товаров, посчитать общую сумму покупки клиента и учесть скидки. Если клиент делает покупки на сумму более от 1000с., но менее 3000с., то скидка составляет 3%, если от 3000с. но менее 6000с., то скидка составляет 5%, если покупка на сумму более 6000с., то скидка - 10%.

#### **Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты.
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Провести автоматический анализ кода
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

#### **Спецификация задачи №П.17**

Создать систему для учета продаж авиабилетов. Система должна обеспечить ввод и хранение данных по рейсам (тип самолета, вместимость, номер рейса, время и день вылета, пункт назначения, стоимость билета для взрослого пассажира в эконом-классе и бизнес - классе), по пассажирам (Ф.И.О., номер паспорта). Для детей скидка 50%. По запросу пользователя система должна выдать сколько мест продано на заданный рейс, какова сумма продаж за неделю.

#### **Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты.
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

#### **Спецификация задачи №П.18**

Создать систему для учета продаж железнодорожных билетов.

Система должна обеспечить ввод и хранение данных по поездам: тип вагона (плацкарт, купейный), количество вагонов, вместимость вагонов, номер поезда, время и день отправления, пункт назначения, стоимость билета для взрослого пассажира), по пассажирам (Ф.И.О., номер паспорта). Для детей скидка 50%. По запросу пользователя система должна выдать сколько мест продано на заданный поезд, какова сумма продаж за неделю.

**Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты.
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### **Спецификация задачи №П.19**

Создать ЭЛЕКТРОННЫЙ ТЕЛЕФОННЫЙ СПРАВОЧНИК.

Составить программу, которая позволит вводить и сохранять телефонные номера с указанием имени абонента, его адреса, категории (друзья, родственники, мастерские, магазины и т.д.), дополнительные данные. По запросу пользователя программа должна находить телеф. номер по любому ключу.

**Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты.
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);
5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Провести автоматический анализ кода
7. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### **Спецификация задачи №П.20**

Создать систему для учета возвратов книг в библиотеку.

Система фиксирует информацию о читателях – ФИО, контактная информация, выданные книги, даты выдачи и возврата книг. Контролируя значение поля с датой возврата книги, необходимо вести учет должников. Отчеты должны показывать список всех выданных на руки книг и журналов.

**Требуется:**

1. Разработать спецификацию требований к ПО.  
Построить диаграмму вариантов использования.
2. Построить план тестирования. Разработать тестовые варианты.
3. Спроектировать ПО (GUI, структура БД, построить диаграммы деятельности, классов, пакетов, последовательности);
4. Сконструировать рабочую версию ПО (выбрать язык программирования, инструментальную среду, написать программу, провести модульное тестирование);

5. Провести тестирование интерфейса, системное тестирование, профилирование программы;
6. Валидация ПО (показать, что ПО удовлетворяет требованиям stakeholders).

### Спецификация задачи №III.1

#### Система электронного голосования

Написать программу для проведения выборов. Избиратели должны предварительно зарегистрироваться в системе. Администратор заполняет список кандидатов. Каждый участник (по сети) или с того же самого компьютера входит в систему и голосует. Данные о проголосовавших накапливаются в базе данных. Дважды проголосовать нельзя. По окончании периода голосования администратор запускает процедуру подсчета голосов, и система выдает результат

#### Требуется:

5. Разработать спецификацию требований к ПО, которое решает эту проблему;
6. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
7. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
8. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи № III.2

#### Обработка заказов на доставку пиццы

Написать программу для составления оптимального маршрута разносчика пиццы. Пользователи, желающие заказать пиццу, оставляют в системе заявку, указывая район своего проживания. Кроме того, система постоянно накапливает информацию о способах передвижения между соседними районами города. Когда поступает заказ, программа должна найти короткий путь от текущего местонахождения вагончика с пиццей до района проживания заказчика.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders)

### Спецификация задачи № III.3

#### Система «Ипподром»

Написать программу для приема ставок и расчета выигрышей на скачках. Пользователи, зарегистрированные в системе, могут поставить любую сумму на одну из семи лошадей. Предварительно администратор системы вносит список лошадей. По результатам заезда (его можно смоделировать на основе случайных чисел) должна рассчитываться сумма выигрыша.

Желательно, чтобы система могла обрабатывать несколько видов ставок (какая лошадь придет первой, какая лошадь придет последней и т.д.).

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;

2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders)

#### **Спецификация задачи № III.4**

##### **Система контроля за пробками**

Написать программу для автомобилистов, отслеживающую возникающие на дорогах пробки и предлагающую наименее загруженный путь между двумя точками. Автомобилисты могут использовать эту программу, чтобы сообщить о пробке на пути от одной точки города до соседней (этими точками могут быть смежные районы города). Каждая пробка получает оценку загруженности по десятибалльной системе. Программа также по запросу для двух точек находит маршрут, на котором общая продолжительность стояния в пробках минимальна (разницу в расстоянии между разными смежными районами мы игнорируем).

##### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № III.5**

##### **Система «Складской учет».**

Написать программу для заведующего склада, отслеживающую приход и расход товара. Система должна показывать список хранящихся на складе товаров и их количество, остаточную стоимость и оборотную ведомость.

##### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № III.6**

##### **Бизнес-аукцион**

Написать программу для проведения торгов по заявкам. Участники регистрируются в системе и подают заявки на продажу и закупку товара (кроме наименования и цены товара могут указываться дополнительные параметры, такие как качество). В конце дня система (по команде администратора) сопоставляет заявки и выдает результаты торгов, пытаясь максимально удовлетворить потребности участников. Заявки, которые остались неудовлетворены, переносятся на следующий день.

##### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### **Спецификация задачи № III.7**

#### **Система для координации участников проекта**

Необходимо разработать программу для координации группы участников проекта.

Участникам проекта необходимо назначить время и место встречи. Каждый участник регистрируется в системе и указывает свои возможности (список промежутков времени и мест, где он может встретиться). Система подводит итоги, определяя возможные варианты времени и места встречи, подходящие для всех участников.

Возможное время встречи должно задаваться сочетанием день недели + с какого часа + до какого часа. Место встречи выбирается из списка, который заполняется заранее.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### **Спецификация задачи № III.8**

#### **Система «Регистрация больных в поликлинике».**

Написать программу для учета больных в поликлинике. Система должна регистрировать больных в базе данных, выводить полный список зарегистрированных больных и осуществлять поиск по базе данных. Также система должна иметь базу данных с врачами, работающими в данной поликлинике и назначать каждому пациенту врача.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### **Спецификация задачи № III.9**

#### **Система распределения вакансий на рынке труда**

Написать программу для распределения вакансий. Пользователи (работники и работодатели) регистрируются в системе, после чего могут делать заявки. Можно предложить работу (указав требования к работнику, зарплату и другие параметры), либо искать работу (указать желаемую должность, зарплату и т.д.). После накопления достаточного количества заявок система приступает к их распределению, чтобы максимально учесть пожелания пользователей и возвращает результат в виде пар работник-работодатель.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

**Спецификация задачи № III.10****Система тестирования**

Написать программу для проведения тестирования. Каждый преподаватель должен иметь возможность добавлять в программу вопросы (указывая при этом варианты ответов и отмечая правильный вариант). Пользователи проходят тестирование, отвечая на эти вопросы и получая оценку в виде процента правильных ответов.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

**Спецификация задачи № IV.1**

Написать метод вычисления факториала числа, результат вычислений передавать в выходном параметре. Если метод отработал успешно, то вернуть значение true; если в процессе вычисления возникло переполнение, то вернуть значение false.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);

Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders)

**Спецификация задачи № IV.2**

Написать программу, которая вычисляет число гласных и согласных букв в файле. Имя файла передавать как аргумент в функцию Main. Содержимое текстового файла заносится в массив символов. Количество гласных и согласных букв определяется проходом по массиву. Предусмотреть метод, входным параметром которого является массив символов из файла. Метод вычисляет количество гласных и согласных букв.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

**Спецификация задачи № IV.3**

Написать программу, вычисляющую среднюю температуру за год из набора, введенного пользователем. Значения температур за каждый месяц необходимо сохранить в одномерный массив. Написать функцию, которая по данному массиву вычисляет среднюю температуру и возвращает среднее значение.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

**Спецификация задачи № IV.4**

Создать двумерный массив `temperature [12,30]`, в котором будет храниться температура для каждого дня месяца (предполагается, что в каждом месяце 30 дней). Сгенерировать значения температур случайным образом. Для каждого месяца распечатать среднюю температуру. Для этого написать метод, который по массиву `temperature [12,30]` для каждого месяца вычисляет среднюю температуру в нем, и в качестве результата возвращает массив средних температур. Полученный массив средних температур отсортировать по возрастанию.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

**Спецификация задачи № IV.5**

Наибольший общий делитель (НОД) двух целых чисел  $x$  и  $y$  — это наибольшее целое, на которое без остатка делится каждое из двух чисел. Напишите рекурсивную функцию `nod`, которая возвращает наибольший общий делитель чисел  $x$  и  $y$ . НОД для  $x$  и  $y$  определяется рекурсивно следующим образом: если  $y$  равно 0, то `nod(x, y)` возвращает  $x$ ; в противном случае `nod(x, y)` равняется `nod(y, x % y)`, где `%` — это операция вычисления остатка.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.6**

Написать программу, которая решает квадратное уравнение. Входные данные – коэффициенты уравнения, выходные – найденные корни.

##### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.7**

Написать программу, которой на вход подается два целых числа, на выходе – результат деления одного числа на другое. Предусмотреть обработку исключительной ситуации, возникающей при делении числа на ноль.

##### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.8**

Написать программу, которая вводит число от 1 до 365 (номер дня в году), переводит это число в месяц и день месяца. Например, число 40 соответствует 9 февраля (високосный год не учитывать).

##### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи № IV.9

Написать программу, реализующую умножение двух матриц. В программе предусмотреть два метода: метод печати матрицы, метод умножения матриц (на вход две матрицы, возвращаемое значение – матрица).

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи № IV.10

Реализовать класс для описания здания (уникальный номер здания, высота, этажность, количество квартир, подъездов). Поля сделать закрытыми, предусмотреть методы для заполнения полей и получения значений полей для печати. Добавить методы вычисления высоты этажа, количества квартир в подъезде, количества квартир на этаже и т.д. Предусмотреть возможность, чтобы уникальный номер здания генерировался программно. Для этого в классе предусмотреть статическое поле, которое бы хранило последний использованный номер здания, и предусмотреть метод, который увеличивал бы значение этого поля.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи № IV.11

Подсчитать сумму десятичных цифр введенного натурального числа.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

### Спецификация задачи № IV.12

Даны количество  $N$  подряд идущих нечетных чисел и их сумма  $S$ . Если  $N$  и  $S$  заданы правильно, то выведите этот ряд нечетных чисел.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;

2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.13**

Дан текст. Определить, содержит ли он символы, отличные от букв и пробела.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.14**

Дано натуральное число  $n$  ( $n \leq 100$ ), определяющее возраст человека (в годах). Дать для этого числа наименования «год», «года», или «лет». Например, 1 год, 23 года, 45 лет и т.д.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.15**

Дан одномерный массив. Написать функцию, определяющую минимальный, максимальный элементы массива и среднее арифметическое элементов. Кроме того, программа должна иметь функцию ввода одномерного массива и функцию вывода.

#### **Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.16**

Написать функцию перемножения матриц  $A$  размером  $n \times m$  и  $B$  размером  $m \times l$ . Элементы результирующей матрицы получить с помощью следующей формулы. Массивы должны быть динамическими.

$$C_{ik} = \sum_{j=1}^m A_{ij} B_{jk}$$

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.17**

Написать функции вычисления суммы элементов каждой строки матрицы  $A$  размером  $n \times m$ , определения наибольшего значения этих сумм.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.18**

Дана действительная матрица размера  $6 \times 9$ . Найти среднее арифметическое наибольшего и наименьшего значений ее элементов. Программа должна быть составлена с использованием функций.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### **Спецификация задачи № IV.19**

В квадратной матрице размера  $n \times n$  найти значение наибольшего по модулю элемента матрицы, а также определить индексы этого элемента. Предполагается, что такой элемент - единственный. Программа должна быть составлена с использованием функций.

**Требуется:**

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);

3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### Спецификация задачи № IV.20

Напишите программу, которая использует шаблон функции `maximum` для поиска максимального из трех целых чисел, трех чисел с плавающей запятой и трех символов.

#### Требуется:

1. Разработать спецификацию требований к ПО, которое решает эту проблему;
2. Спроектировать ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты);
3. Сконструировать ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО);
4. Продемонстрировать валидность программы (показать, что программа удовлетворяет требованиям stakeholders).

#### Спецификация задачи № V.I

Имеется справочник видов сырья, в котором отражены: код, наименование, единица измерения, количество, сумма (денежная); справочник единиц измерения; протокол закупок: дата, сырье, количество, сумма; а также таблица (например, *Finance*), отражающая текущую сумму финансов. Исследуемая предметная область – закупка сырья.

#### Требуется:

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) При новой поставке сырья в протокол закупок заносится новая запись с указанием даты поставки, вида сырья, поставляемого количества и суммы (общей стоимости купленного сырья);
  - б) При этом автоматически уменьшается имеющаяся сумма в таблице *Finance*, а также увеличивается количество и сумма по соответствующему сырью в справочнике сырья;
  - в) Перед выполнением транзакции необходимо осуществлять проверку на существование соответствующей суммы в таблице *Finance*; при отсутствии требуемой суммы необходимо выдать сообщение о нехватке ресурсов и прекратить выполнение процедуры.
3. Создать объекты БД для решения поставленной задачи (виды сырья выбрать на свое усмотрение).
4. Предусмотреть целостность данных для каждой из таблиц.
5. Протестировать работу системы на нескольких примерах и показать корректность решения.
6. Обосновать эффективность выбранного пути решения.

#### Спецификация задачи № V.II

Имеется справочник видов сырья, в котором отражены: код, наименование, единица измерения, количество; справочник готовой продукции: код, наименование, единица

измерения, количество; справочник ингредиентов: продукция, сырье, количество; справочник единиц измерения; протокол производства: дата, продукция, количество. Исследуемая предметная область – производство продукции.

**Требуется:**

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) При новом производстве в протокол производства заносится новая запись с указанием даты, вида продукции, вида сырья, производимого количества;
  - б) При этом автоматически уменьшается имеющееся количество в справочнике сырья, а также увеличивается количество соответствующего продукта в справочнике продукции;
  - в) Перед выполнением транзакции необходимо осуществлять проверку на существование соответствующего количества сырья для требуемого объема производимой продукции; при отсутствии требуемого количества необходимо выдать сообщение о нехватке ресурсов и прекратить выполнение процедуры.
3. Создать объекты БД для решения поставленной задачи (виды сырья и продукции выбрать на свое усмотрение).
4. Предусмотреть целостность данных для каждой из таблиц.
5. Протестировать работу системы на нескольких примерах и показать корректность решения.
6. Обосновать эффективность выбранного пути решения.

**Спецификация задачи № V.III**

Имеется справочник видов продукции, в котором отражены: код, наименование, единица измерения, количество; справочник единиц измерения; протокол продаж: дата, продукция, количество, цена; а также таблица (например, *Finance*), отражающая текущую сумму финансов. Исследуемая предметная область – продажа продукции.

**Требуется:**

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) При новой продаже продукции в протокол продаж заносится новая запись с указанием даты продажи, вида продукции, продаваемого количества и цены за единицу продукции;
  - б) При этом автоматически увеличивается имеющаяся сумма в таблице *Finance*, а также уменьшается количество соответствующего продукта в справочнике продукции;
  - в) Перед выполнением транзакции необходимо осуществлять проверку на существование соответствующего количества продукции; при отсутствии требуемого количества необходимо выдать сообщение о нехватке ресурсов и прекратить выполнение процедуры.
3. Создать объекты БД для решения поставленной задачи (виды продукции выбрать на свое усмотрение).
4. Предусмотреть целостность данных для каждой из таблиц.
5. Протестировать работу системы на нескольких примерах и показать корректность решения.
6. Обосновать эффективность выбранного пути решения.

**Спецификация задачи № V.IV**

Имеется справочник сотрудников, в котором отражены: ID, фамилия, имя, отчество, дата рождения, образование, адрес, телефон; справочники должностей, образования (среднее, средне-профессиональное, незаконченное высшее, высшее); справочник управлений, отделов, которые входят в управления; деятельность сотрудников: сотрудник, отдел, должность, дата приема, дата увольнения. Причем, если дата увольнения остается открытой, то сотрудник в настоящее время работает в этой должности. Исследуемая предметная область – деятельность сотрудников.

**Требуется:**

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) При поступлении на работу вводятся данные нового сотрудника, дата поступления, должность, отдел, при этом дата увольнения остается пустой;
  - б) При переводе с должности на должность проставляется дата, которая автоматически ставится как дата принятия на новую и дата увольнения со старой должности.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение, которое будет позволять просматривать сотрудников выбранного управления, отдела, а также историю деятельности выбранного сотрудника.
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

**Спецификация задачи № V.V**

Имеется справочник жителей, в котором отражены: ID, фамилия, имя, отчество, дата рождения, ИНН, адрес, телефон; справочник областей; справочник районов, которые входят в области. Идентификация жителей проводится по ИНН (14-значая цифровая величина), в связи с чем предусмотреть запрет повторного его ввода. Исследуемая предметная область – учет жителей республики.

**Требуется:**

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) При вводе нового жителя проверяется отсутствие его ИНН. Если в базе такого номера нет, то ввод разрешается, в обратном случае выводится сообщение, что такой номер имеется, и ввод запрещается;
  - б) Разрешается перемещаться внутри республики, при этом выбирается новая область, район и наименование населенного пункта;
  - в) Предусмотреть возможность первый раз вводить название улицы, а затем выбирать ее из списка.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение.
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

## Спецификация задачи № V.VI

Имеется справочник сотрудников, в котором отражены: ID, фамилия, имя, отчество, дата рождения, адрес, телефон; справочники должностей, управлений, отделов, которые входят в управления; должности сотрудников: сотрудник, отдел, должность; деятельность сотрудников: дата, сотрудник, сумма прихода; начисляемый процент: дата начала, процент, дата закрытия. Каждому сотруднику выдается зарплата с начислением процента от общей суммы прихода, совершенной этим сотрудником. Исследуемая предметная область – начисление зарплат сотрудников.

### Требуется:

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) Зарплату сотруднику выдается с учетом суммы прихода за данный месяц, приведённой сотрудником, а именно определенный процент, который проставляется в отдельной таблице;
  - б) Причем, если дата закрытия остается открытой, то в настоящее время принимается процент, указанный в этой записи.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение, которое будет позволять просматривать выданную зарплату сотрудников выбранного управления, отдела.
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

## Спецификация задачи № V.VII

Имеется справочник спортивных команд, в котором отражены: ID, наименование, очки; спортивные состязания; дата встречи, 1-я команда, 2-я команда, голы 1-й команды, голы 2-й команды. После каждой встречи победитель получает 3, проигравший – 0 очков, при ничейном исходе обе команды получают по 1 очку. Исследуемая предметная область – спортивные состязания.

### Требуется:

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) При окончании матча в таблицу заносится результат матча;
  - б) Причем, автоматически для каждой команды изменяется количество очков в зависимости от исхода матча.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение, которое будет позволять просматривать общий итог чемпионата, историю игр выбранной команды, а также занятые места по районам.
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

## Спецификация задачи № V.VIII

Имеется справочник товаров, в котором отражены: ID, наименование, единица измерения; справочник единиц измерения; таблица прихода товаров: дата, товар, количество, поставщик; таблица выдачи товаров: дата, товар, количество, заказчик; справочники поставщиков и заказчиков. Исследуемая предметная область – розничный склад.

### Требуется:

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) Наличие товара вычисляется по приходу на склад и выдаче товара со склада;
  - б) Причем, при выдаче со склада проводится проверка наличия требуемого количества товара, и транзакция проводится только при положительном ответе, в противном случае выдается соответствующее сообщение, и операция не производится.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение, которое будет позволять просматривать наличие товаров, а также приход-расход товаров за выбранный период.
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

## Спецификация задачи № V.IX

Имеется справочник пациентов, в котором отражены: ID, фамилия, имя, отчество, дата рождения, адрес, телефон; справочник заболеваний; справочник медицинских манипуляций; таблица пролеченных случаев: пациент, дата госпитализации, диагноз (по справочнику заболеваний), дата выписки; таблица манипуляций: пациент, дата, манипуляция. Исследуемая предметная область – учет деятельности стационара.

### Требуется:

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) Медицинские манипуляции относятся к тому случаю госпитализации, для которого дата манипуляции лежит между датой госпитализации и датой выписки;
  - б) Один и тот же пациент может госпитализироваться несколько раз, в связи с чем при повторном случае данные пациента не вносятся, а выбираются из базы.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение, предоставляющее возможность просматривать историю госпитализации, лечения выбранного больного, а также получать сведения о количестве пролеченных больных за определенный период.
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

## Спецификация задачи № V.X

Имеется справочник членов семьи, в котором отражены: ID, имя, дата рождения, членство; справочник членства (дедушка, бабушка, отец, мать, сын, дочь); справочник статей расхода-прихода, в котором имеется признак (расход либо приход); таблица операций: дата, член семьи, статья, сумма. Исследуемая предметная область – учет финансовых средств семьи.

### Требуется:

1. Разработать базу данных на SQL SERVER с созданием перечисленных выше таблиц с определением отношений между ними.
2. Выявить пути решения поставленной задачи:
  - а) Учет финансовых средств осуществляется за счет прибавления в случае прихода и вычитания в случае расхода;
  - б) При добавлении операции на расход требуется проверка на наличие требуемой суммы. При отсутствии требуемой суммы необходимо выдать сообщение о нехватке ресурсов и прекратить выполнение процедуры.
3. Создать объекты БД для решения поставленной задачи.
4. Предусмотреть целостность данных для каждой из таблиц.
5. С самостоятельным выбором платформы разработать клиентское приложение, предоставляющее возможность просматривать историю финансовых операций для каждого члена семьи или в целом семьи за определенный период с вычислением баланса (сумма прихода минус сумма расхода).
6. Протестировать работу системы на нескольких примерах и показать корректность решения.
7. Обосновать эффективность выбранного пути решения.

## Спецификация задачи № VI.1

В здании аэропорта на экранах отображается информация о самолетах, а именно: информация о пункте отправления, пункте назначения, номере рейса, времени прибытия, времени отправления, номере секции для регистрации. Экраны – это средство, которое помогает своевременно зарегистрировать и отправить пассажиров. Важно, чтобы информация на экранах была понятной и верной.

Создайте необходимую информацию в виде таблицы для такого экрана, с помощью класса Aeroflot, содержащего в описании следующие поля:

- номер рейса;
- название пункта отправления;
- название пункта назначения рейса;
- время прибытия;
- время отправления;
- место регистрации.

Напишите код программы, выполняющей следующие действия:

- ввод с клавиатуры значений полей объектов;
- сортировку записей в таблице в алфавитном порядке по названию пунктов назначения;
- вывод на консоль значений полей объектов класса в виде таблицы рейсов;
- если таких рейсов нет, выдать соответствующее сообщение.

### Задания:

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке объектно-ориентированного программирования C++, протестируйте и отладьте его.

### Спецификация задачи № VI.2

Для участия в ежегодном трансконтинентальном Ралли-марафоне «Дакар» по усложненной трассе в Южной Америке выбраны грузовики КАМАЗ и Tatra, которые после 2000 года лидируют в зачете грузовиков.

Решите задачу сравнения скоростей движения грузовиков по разным по проходимости участкам трассы, а именно: по равнине, горам, пустыне. Создайте и выдайте на экран таблицу результатов ралли - марафона. Определите победителя.

Для решения задачи используйте классы **Kamaz** и **Tatra**, а также функцию сравнения скоростей **FrCreator** (). Функция **FrCreator** () возвращает число +1, если объект **kamaz** движется быстрее объекта **tatra**; нуль, если их скорости одинаковы; число -1, если объект **kamaz** движется медленнее объекта **tatra**.

Оба класса содержат поля: «скорость» и «наименование» грузовой машины, а также методы: инициализация и отображение полей на экране. Определитесь с идентификаторами доступа к членам класса, не нарушая принципа инкапсуляции.

Сделайте функцию **FrCreator()** дружественной: классу **Kamaz**, обоим классам **Kamaz** и **Tatra**.

Сделайте класс **Tatra** дружественным классу **Kamaz**.

### Задания:

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке объектно-ориентированного программирования C++, протестируйте и отладьте его.

### Спецификация задачи № VI.3

Типы данных полезны там, где ошибки могут быть вызваны арифметическим переполнением, которое не допустимо.

Создайте и откомпилируйте класс **Int**. Перегрузите четыре бинарных целочисленных арифметических операции (+, -, \*, /) и унарные операции постфиксной и префиксной

форм инкремента с помощью внутренней операторной функции так, чтобы их можно было использовать для операций с объектами класса **Int**.

Если результат какой-либо из них выходит за границы типа **int** (в 32-битной системе), имеющее значения от **2 147 483 648** до **-2 147 483 648**, то операция должна послать сообщение об ошибке и завершить программу. Для выявления ошибки арифметического переполнения используйте концепцию **исключения**.

Для облегчения проверки переполнения выполняйте вычисления с использованием типа **long double**. При описании унарных операций используйте указатель **this**.

### Задания:

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП C++, протестируйте и отладьте его.

### Спецификация задачи № VI.4

Проблему экономии времени, доходов, а также увеличения надежности создаваемой объектно-ориентированной программы можно решить с помощью одного из главных принципов ООП - наследования, используя для создания класса-потомка готовый протестированный и отлаженный код класса-родителя.

Для решения этой проблемы используйте преимущества **простого наследования**. Для этого создайте родительский класс работник (**Employee**), протестируйте и отладьте его. В описании этого класса есть

1. поля:
  - имя работника - **name [64]**,
  - занимаемая должность - **position [64]**,
  - заработная плата за месяц- **salary**;
2. методы:
  - конструктор по умолчанию,
  - конструктор с аргументами,
  - деструктор,
  - расчет заработной платы за год,
  - размещение информации о классе-родителе на консоли.

На основе отлаженного класса-родителя **Employee** создайте с помощью **механизма наследования public** класс-потомок менеджер (**Manager**).

Класс - потомок **Manager** состоит из:

1. полей:
  - ежегодный бонус (**annual\_bonus**),
  - машина компании (**company\_car [64]**),
  - опционы на акции (**stock\_options**) типа (**int**).
2. методов:
  - конструктор по умолчанию,

- конструктор с аргументами,
- деструктор,
- размещение информации о классе-потомке на консоли.

*Определитесь с идентификаторами доступа к членам класса, не нарушая основного принципа ООП – инкапсуляции.*

#### **Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту проблему.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП C++, протестируйте и отладьте его.

### **Спецификация задачи № VI.5**

Проблему экономии времени, доходов, а также увеличения надежности объектно-ориентированной программы можно решить с помощью одного из главного принципа ООП - наследования, используя для создания класса-потомка готовые протестированные и отлаженные коды классов-родителей.

Для решения этой проблемы используйте преимущества **множественного наследования**, а именно: **разработайте**, протестируйте и отладьте родительские классы Дисплей (*Display*) и Материнская плата (*MotherBoard*), и на их основе создайте класс-потомок Компьютер (*Computer*):

В описании класса-родителя *Display* есть

1. поля:
  - `char type[32];` //тип монитора
  - `long colors;` // количество цветов
  - `int x_resolution;` // разрешение по оси x
  - `int y_resolution` // разрешение по оси y
2. методы:
  - конструктор с аргументами;
  - размещение информации о классе-родителе на консоли.

В описании класса - родителя *MotherBoard* имеет

1. поля:
  - `int processor;` // тип процессора
  - `int speed;` //скорость процессора
  - `int RAM;` // объем оперативной памяти
2. методы:
  - конструктор с аргументами;
  - размещение информации о классе-родителе на консоли.

- Класс-потомок *Computer* создать на основе родительских классов с помощью механизма наследования **public** и отобразить его на экране, добавив:

- 1) поля:
  - марка компьютера (**name [64]**),
  - объем жесткого диска (**hard\_disk**).
- 2) методы:
  - Конструктор с аргументами,

- метод Show ().

**Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту проблему.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП С++, протестируйте и отладьте его.

### Спецификация задачи № VI.6

Описать класс, реализующий стек (**Stack**). Разработать программу, использующую этот класс для моделирования Т-образного сортировочного узла на железной дороге.

Программа должна разделять на два направления состав, состоящий из вагонов двух типов (на каждое направление формируется состав из вагонов одного типа). Предусмотреть возможность формирования состава из файла или с клавиатуры.

Возможно использование стандартных функций при работе со стеком в виде контейнера из библиотеки STL:

- **push()** - поместить элемент в вершину стека ;
- **pop()** - удалить элемент из вершины стека;
- **printStack()** - вывести стек на экран;
- **getStackSize()** – вернуть размер стека.

**Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП С++, протестируйте и отладьте его.

### Спецификация задачи № VI.7

Создать **шаблон** класса «стек» (**Stack**). Разработать программу, использующую этот шаблон класса для моделирования Т-образного сортировочного узла на железной дороге.

Программа должна разделять на два направления состав, состоящий из вагонов двух типов (на каждое направление формируется состав из вагонов одного типа). Предусмотреть возможность формирования состава из файла и с клавиатуры.

Возможно использование стандартных функций при работе со стеком в виде контейнера из библиотеки STL:

- **push()** - поместить элемент в вершину стека ;
- **pop()** - удалить элемент из вершины стека;
- **printStack()** - вывести стек на экран;
- **getStackSize()** – вернуть размер стека.

**Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП C++, протестируйте и отладьте его.

### Спецификация задачи № VI.8

Опишите класс «студенческая группа».

Предусмотрите возможность:

1. работы с переменным числом студентов;
2. поиска студента по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона);
3. добавления или удаления записей;
4. сортировки по разным полям.

Разработайте программу, демонстрирующую работу с этим классом.

Программа должна содержать меню, позволяющее осуществить проверку всех методов класса.

**Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП C++, протестируйте и отладьте его.

### Спецификация задачи № VI.9

Опишите класс «записная книжка».

Предусмотрите возможность:

1. работы с произвольным числом записей;
2. поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона);
3. добавления или удаления записей;
4. сортировки по полю «фамилия».

Разработайте программу, демонстрирующую работу с этим классом.

Программа должна содержать меню, позволяющее осуществить проверку всех методов класса.

**Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.

3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП С++, протестируйте и отладьте его.

### Спецификация задачи № VI.10

Оптимизируйте расчет площадей разных геометрических фигур на основе одного линейного размера с помощью виртуальной функции при наследовании классов.

Для этого разработайте программу с абстрактным классом фигура (**Figure**), в котором есть открытое числовое поле **size** (линейный размер геометрической фигуры), конструктор, который полю **size** присваивается единичное значение и чисто виртуальный методом **area()** для вычисления площади фигуры.

На основе этого абстрактного класса создайте три производных класса с помощью механизма наследования **public**. В каждом производном классе переопределите метод **area()** более подходящим образом для этого производного класса. А именно: в классе **Circle** метод **area()** переопределите для вычисления площади круга, в классе **Square** метод **area()** переопределите для вычисления площади квадрата, а в классе **Triangle** метод **area()** переопределите для вычисления площади равностороннего треугольника по формуле  $\sqrt{3} * size * size / 4$ . Число  $\pi$  определите, как глобальную константную переменную.

Для решения задачи рекомендуется использовать *переопределение* чисто виртуальной функции базового класса в производных классах.

**Задания:**

1. Разработайте спецификацию требований к ПО, которая решает эту задачу.
2. Спроектируйте ПО: входной и выходной интерфейсы для консоли, блок-схему программы.
3. Определитесь со структурой программы (прототипы, описание, вызов функций), разработайте код на языке ООП С++, протестируйте и отладьте

### 4. Критерии оценки знаний студентов

Экзамен проводится в форме устного экзамена с применением компьютера. Для экзамена ГАК специально готовятся экзаменационные билеты, с помощью которых проверяются знания и практические навыки программирования. В билетах приводятся спецификации задач и моделируется реальная ситуация при найме выпускников на работу в software company, когда у соискателей-программистов оценивают умение и практические навыки разрабатывать программный проект за определенный срок. Экзаменационный билет должен включать задание на разработку ПО для решения конкретной прикладной проблемы. (экзаменационные билеты спецификациями задач должны иметь трудоемкость по продолжительности времени разработки программы не более 80–120 минут.)

Оценка «отлично» (87-100 баллов) выставляется студенту, который правильно разработает спецификацию требований к ПО, спроектирует ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты), сконструирует ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО) и продемонстрирует валидность программы (показать, что программа удовлетворяет требованиям stakeholders). Студент должен показать всестороннее диалектико-

материалистическое понимание основных вопросов ПИ, уметь грамотно изложить результаты, проделанной работы по выполнению задания, показавшему, ответить четко и по существу на все вопросы комиссии.

Оценка «хорошо» (74-86 баллов) выставляется студенту, который правильно разработает спецификацию требований к ПО, спроектирует ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты), сконструирует ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО) и продемонстрирует валидность программы (показать, что программа удовлетворяет требованиям stakeholders). Студент должен уметь, грамотно изложить результаты, проделанной работы по выполнению задания, показавшему, ответить четко и по существу на все вопросы комиссии.

Оценка «удовлетворительно» (61-73 балла) ставится студенту, который разработает спецификацию требований к ПО, спроектирует ПО (входной, выходной интерфейсы, блок-схемы, диаграммы потоков данных, архитектура ПО, тесты), сконструирует ПО (определить структуру модулей ПО, выбрать язык программирования – инструментальное средство, написать программу – кодировать модули ПО). Доказывая его способность к самостоятельно допускающему неточности в ответе, но обладающему необходимыми знаниями для их устранения при помощи экзаменатора, набравшему необходимую минимальную сумму баллов.

Оценка «неудовлетворительно» (менее 61 балла) ставится студенту, обнаружившему пробелы в необходимом объеме знаний по дисциплине, допустившему принципиальные ошибки в ответе, не сумевшему конкретизировать ответы, привести практически примеры.