Критерии оценивания заданий заключительного этапа по направлению «Компьютерные системы и сети»

Задания по направлению состояли только из инвариантной части. Для того, чтобы претендовать на статусы медалиста, дипломанта I, II, III степени, участникам необходимо набрать наибольшее число баллов за все задания.

Номер и тематика задания	Максимальный балл	Учёт в рейтинге по направлению
1. Схемотехника	20	✓
2. Базы данных	20	✓
3. Вычислительные системы	20	✓
4. Технологии программирования	20	✓
5. Компьютерные сети	20	✓

ЗАДАЧА № 1. Схемотехника

Задание

Система электрического питания на основе фотоэлектрических панелей оснащена трекерами, обеспечивающими мониторинг перемещения солнца и корректировки положения платформы с фотоэлектрическими панелями вслед за солнцем. С помощью трех сервоприводов реализуется оптимальное позиционирование платформы посредством трех видов перемещения/вращения: перемещение по вертикали, повороты вверх-вниз и повороты влево-вправо. Каждому виду перемещения/вращения соответствует отдельный сервопривод. Четырехчасовой цикл работы сервоприводов описывается следующим образом:

Первый час – платформа перемещается по вертикали, поворачивается вверх-вниз и влево-вправо.

Второй час – платформа поворачивается только вверх-вниз, перемещение по вертикали и поворот влево-вправо не осуществляются.

Третий час – платформа поворачивается только влево-вправо, перемещение по вертикали и поворот вверх-вниз не осуществляются.

Четвертый час – платформа поворачивается влево-вправо и вверх-вниз, по вертикали перемещение не осуществляется.

Цикл начинается с первого часа и после окончания четырехчасового цикла, цикл начинается заново с первого часа.

Сервопривод перемещает/поворачивает платформу на заданное разработчиком в алгоритме значение, когда на вход сервопривода подан сигнал логическая «1», и не перемещает/не вращает платформу, когда на вход сервопривода подан сигнал логический «0». В начале каждого часа по срабатыванию тактового сигнала CLK (переключение из «0» в «1» и обратно) происходит переключение режима работы согласно описанию выше.

Задание: составить схему управления сервоприводами на основе логических элементов и/или JK-/D-триггеров с синхронным управлением. В качестве входного сигнала схемы используется тактовый сигнал CLK в форме меандра. Изменение сигнала на выходе у JK-триггера происходит по заднему фронту тактового сигнала, у D-триггера по переднему фронту тактового сигнала. Сами сервоприводы достаточно представить в виде трех отдельных блоков с одним входом у каждого. Привести текстовое описание принципа работы схемы. Начальное условие (состояние выходов триггеров в первый момент времени) указываются текстом, схему сброса реализовывать не требуется.

Задача № 1. Решение

Пример одного из способов решения:

На первом этапе необходимо выявить закономерности в переключении сервоприводов. Из результата анализа закономерностей будет предложена соответствующая схема. На втором этапе из собранных закономерностей реализуется схема управления.

В основе схемы управления предлагается использовать триггеры, в частности D-триггер, по сути, устройство памяти способное запоминать информацию и обновлять ее с поступлением на тактовый вход переднего фронта тактового сигнала. То есть состояние выходов D-триггера определяется комбинацией сигналов на его входе предыдущего такта. Как другой вариант решения использовать логические элементы, формирующие сигнал на выходе сразу же при изменении сигнала на входе, или реализовать счетчик с коэффициентом счета 4 и с логическими элементами для формирования необходимых выходных сигналов.

1. Закономерности для каждого сервопривода.

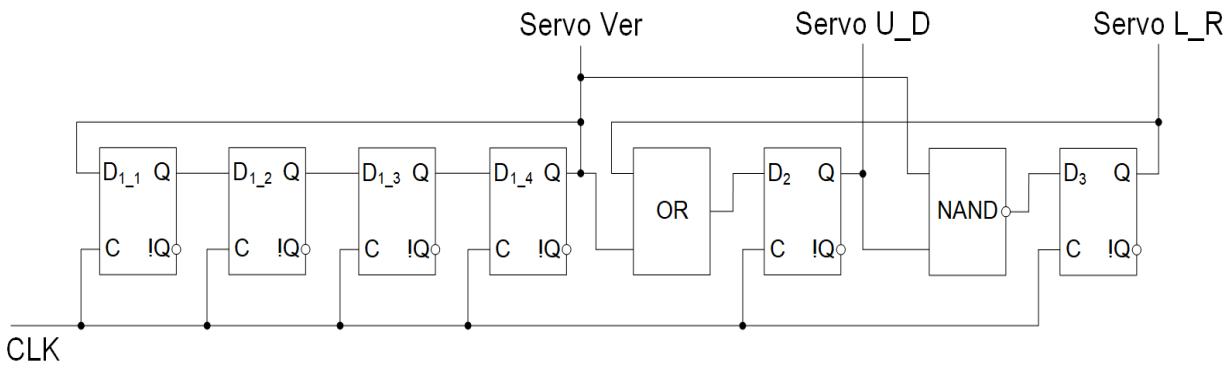
Первый сервопривод – перемещение по вертикали. Первый сервопривод работает в режиме: один цикл HIGH и затем три цикла LOW. Данный режим представляет собой реализацию функции сдвигового регистра с определенным начальными состояниями. Один из способов реализации такого режима это схема на четырех последовательно соединённых D-триггерах.

Второй сервопривод – вращение верх/вниз. Второй сервопривод работает в режиме: два цикла HIGH, один цикл LOW и один цикл HIGH. При рассмотрении системы в целом выявляется закономерность, что в рамках одного цикла состояние второго сервопривода определяется результатом выполнение логической операции ИЛИ для состояний первого и третьего сервоприводов предыдущего такта.

Третий сервопривод – вращение влево/вправо. Третий сервопривод работает в следующем режиме: один цикл HIGH, один LOW и два цикла HIGH. При рассмотрении системы в целом выявляется закономерность, что в рамках одного цикла состояние третьего сервопривода определяется результатом выполнение логической операции И-НЕ для состояний первого и второго сервоприводов предыдущего такта.

2. Схема управления сервоприводами.

Схема, реализующая вышеописанные закономерности, представлена на рисунке ниже.



Все триггеры подключаются к одному тактовому сигналу и переключаются одновременно.

Вход триггера D_{1_1} замкнут на выход Q триггера D_{1_4}, это позволяет реализовать зацикливание выходных состояний триггера на четыре такта. Выход Q триггера D_{1_4} соответствует состоянию сервопривода, реализующего перемещение по вертикали (Servo Ver).

Прямые выходы Q с триггеров D_{1_4} и D_3 подключаются на элемент, реализующий функцию логического сложения (OR). Результат этой операции подается на информационный вход триггера D_2 и записывается на выход Q этого триггера на следующем такте, соответствующего состоянию сервопривода, реализующего вращение верх/вниз (Servo U_D).

Прямые выходы Q с триггеров D_{1_4} и D_2 подключаются на элемент, реализующий функцию логического умножения с инверсией (NAND). Результат этой операции подается на информационный вход триггера D_3 и записывается на выход Q этого триггера на следующем такте, соответствующего состоянию сервопривода, реализующего вращение влево/вправо (Servo L_R).

Необходимо обратить внимание, что для корректной работы схемы перед началом работы необходимо установить на триггерах D_{1_1}, D_{1_2}, D_{1_3}, D_{1_4}, D_2 и D_3 начальные значения. К примеру, для начала работы схемы управления с Первого цикла: выходы триггеров D_{1_4}, D_2 и D_3 устанавливается в значение логической «1». Выходы триггеров D_{1_1}, D_{1_2} и D_{1_3} в значение логического «0».

Критерии оценивания решения задачи № 1

Максимальная оценка за задание – 20 баллов.

Что оценивается	Баллы

А) Составление схемы, реализующей переключения	**10**
• *собранная схема реализует 4 последовательных состояния*	8 баллов
• *собранная схема реализует 3 последовательных состояния*	6 баллов
• *собранная схема реализует 2 последовательных состояния*	3 балла
• *схема реализована на основе JK-/D-триггерах с синхронным управлением (для всех JK-/D-триггеров)*	2 балла
• *в схеме допущены ошибки разной значимости*	–(1÷5) баллов
Б) Приведены пояснения/расчеты/формулы к схеме	**10**
• *приведено верное пояснения принципа работы схемы для 4 состояний*	10 баллов
• *приведено верное пояснения принципа работы схемы для 3 состояний*	8 баллов
• *приведено верное пояснения принципа работы схемы для 2 состояний*	5 баллов
• *в описании допущены ошибки разной значимости*	–(1÷5) баллов

Рекомендуемая литература для подготовки к решению задания № 1

1. Michael Nielsen - Neural Networks and Deep Learning, chapter 2, 2015.

ЗАДАЧА № 2. Базы данных

Задание

Для предметной области "Экскурсионное бюро" приведите к третьей нормальной форме отношение "Экскурсии" (идентификатор (ПК), название достопримечательности, адрес достопримечательности, вид экскурсии, дата и время проведения, продолжительность, ФИО экскурсовода, стоимость экскурсии, максимальное количество участников, количество участников).

Особенности предметной области:

– для каждой достопримечательности может быть несколько видов экскурсий, каждый вид экскурсий может быть для разных достопримечательностей;
– стоимость и продолжительность экскурсии, а также максимальное количество участников зависят от достопримечательности и вида экскурсии;
– стоимость может быть указана за билет (например, 300 руб/чел) или за всю экскурсию (например, 2500 руб/экс);
– адрес достопримечательности хранится в поле символьного типа и не требует нормализации.

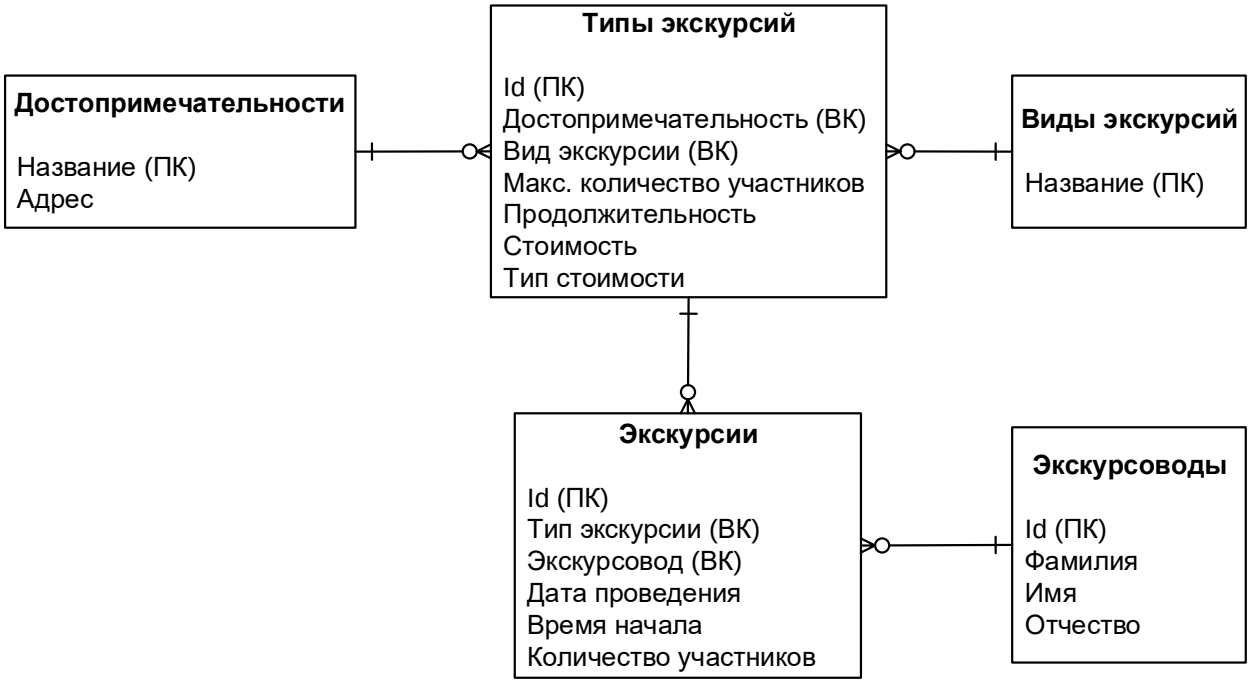
Результат представьте в виде схемы базы данных в одной из общеупотребительных нотаций. Для каждого отношения приведите перечень атрибутов с указанием первичных ключей (ПК) и внешних ключей (ВК).

Напишите на SQL следующие запросы:

1. Достопримечательности, по которым нет экскурсий с максимальным количеством участников более 20-ти.
2. Экскурсии за вчерашний день, которые принесли более 6000 рублей.

Задача № 2. Решение

1) Схема БД:



Пояснения:

а) В таблице "Типы экскурсий" поле "тип стоимости" может принимать одно из двух значений – 'чел' или 'экс'. Если предполагается, что могут появиться дополнительные типы, следует создать справочную таблицу "Типы стоимости", а это поле сделать внешним ключом на нее.

б) В таблице "Типы экскурсий" комбинация внешних ключей уникальна. Можно было бы сделать составной primary key(достопримечательность, вид экскурсии): это потребовало бы создания составного ВК в таблице "Экскурсии", что увеличило бы объем памяти, но сократило бы время на соединение таблиц.

2) Запросы:

1. Достопримечательности, по которым нет экскурсий с максимальным количеством участников более 20-ти:

```
select *
    from Достопримечательности
        where название not in (select достопримечательность
                               from Типы_экскурсий
                               where макс_количество_участников > 20);
```

2. Экскурсии за вчерашний день, которые принесли более 6000 рублей:

```
select э.*
    from Экскурсии э, Типы_экскурсий т
```

```
where э.дата_проведения = current_date -1 and э.тип = т.id
(т.тип_стоимости = 'экс' and т.стоимость > 6000 or
т.тип_стоимости = 'чел' and э.количество_участников*т.стоимость > 6000);
```

Критерии оценивания решения задачи № 2

Оценивается умение проводить нормализацию отношений при проектировании реляционных БД и знание языка SQL. За полностью верное решение ставится 20 баллов. Наличие нижеперечисленных ошибок в ответе снижает оценку следующим образом:

Характер ошибки	Снижение оценки
схема БД (максимальная оценка 12 баллов)	
Неверное определение первичного ключа	на 1 балл за каждую ошибку
Неверное определение внешнего ключа или его отсутствие	на 1 балл за каждую ошибку
Отсутствие указания атрибутов, которые являются первичными и внешними ключами	на 1 балл за каждую ошибку
Объединение любых двух связанных отношений в одно, нарушающее вторую или третью нормальные формы	на 1 балл за каждую ошибку
Необоснованное разбиение любого из отношений на два, разрывающее функциональную зависимость 1:1	на 1 балл за каждую ошибку
Необоснованное разбиение любого из отношений на два, приводящее к дублированию данных	на 1 балл за каждую ошибку
Неверное распределение атрибутов по отношениям	на 1 балл за каждую ошибку
Исчезновение из финальной схемы атрибутов, которые есть в задании	на 1 балл за каждую ошибку
Наличие на схеме связей многие-ко-многим	на 1 балл
Оставление составных атрибутов без изменений	на 1 балл
Наличие на схеме связей, обязательных в обе стороны	на 1 балл
Наличие многозначного внешнего ключа	на 2 балла за каждую ошибку
Появление лишних атрибутов	на 1 балл за каждую ошибку
Наличие вычислимых атрибутов	на 1 балл за каждую ошибку
Неверное изображение внешних ключей	на 1 балл
SQL-запросы (максимальная оценка 8 баллов)	
Неверная логика	на 4 балла в каждом запросе
Неверные условия соединения таблиц или их отсутствие	на 2 балла в каждом запросе
Логическая ошибка	на 2 балла в каждом запросе
Неверные операторы сравнения, функции или предикаты	на 1 балл в каждом запросе
Другие синтаксические ошибки	на 1 балл в каждом запросе

За правильные пояснения к схеме и запросам можно получить дополнительные баллы (не более 2-х).

Рекомендуемая литература для подготовки к решению задания № 2

1. Коннолли Т., Бегг К. Базы данных: проектирование, реализация, сопровождение. Теория и практика, 3-е изд.: Пер. с англ.: Уч. пос.– М.: Изд. дом "Вильямс", 2017. – 1439 с.
   гл. 5. "Язык SQL: манипулирование данными", гл.14. "Методология концептуального проектирования баз данных", гл.15. "Методология логического проектирования баз данных".
2. Карпова И.П. Базы данных. Курс лекций и материалы для практических занятий. – Учебное пособие. – Издательство "Питер", 2013. – 240 с.
   URL: https://publications.hse.ru/books/79801962
   Раздел 3.5. "Извлечение данных из таблиц", глава 9. "Проектирование баз данных".
3. Карпова И. П. Проектирование реляционных баз данных: Методические указания к домашнему заданию по курсу "Базы данных" // М.: МИЭМ НИУ ВШЭ, 2025. – 37 с. URL: http://www.hse.ru/pubs/share/direct/document/1031393910.pdf
4. Изучение основ языка SQL [электронное издание]. – Метод. указания к лабораторным работам 1–4 по курсу "Базы данных" // Сост.: Карпова И.П., Вендин А.С. – Московский институт электроники и математики им. Тихонова НИУ ВШЭ. – М.: 2025. – 35 с. http://www.hse.ru/pubs/share/direct/document/1031391352.pdf

ЗАДАЧА № 3. Вычислительные системы

Задание

На рисунке ниже изображен дамп памяти вычислительной системы на базе процессора с архитектурой Intel 64. Расположите по порядку (сравнив в десятичной системе счисления) три числа: A (single precision floating point), B (short signed integer), C (long unsigned integer), представленных HEX-кодами и расположенных по адресам (см. рис.):

A по адресу 0x0000000000403041,
B по адресу 0x000000000040301D,
C по адресу 0x0000000000403028.

Адрес	Шестнадцатеричное															
0000000000403010	66	6D	65	6D	31	20	69	73	20	64	69	76	69	E4	FF	00
0000000000403020	10	65	20	74	6F	20	66	6D	19	00	00	00	0A	00	66	6D
0000000000403030	65	6D	31	00	00	88	C0	6E	6F	74	20	64	69	76	69	73
0000000000403040	61	62	6C	65	20	74	6F	20	66	6D	65	6D	32	0D	0A	34
0000000000403050	00	FD	01	26	00	11	78	00	6F	04	07	12	00	00	90	00
0000000000403060	F6	FF	FF	FF	F5	FF	FF	FF	00	18	00	25	CC	E1	DE	00

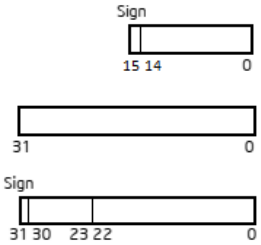
Запишите ответ в виде A=B>C или A<B<C. Объясните свои выводы и ход решения, комментируя: поиск чисел по адресу, определение длины чисел, преобразование согласно формату в десятичное представление, сравнение.

Задача № 3. Решение

Формат single precision floating point (или стандартизированный binary32) предполагает, что число A действительное, со знаком (в старшем бите), имеет размер 4 Байт (32 двоичных разряда) и кодируется согласно стандарту представления дробных чисел в прямом коде.

Формат short signed integer предполагает, что число B – целое со знаком (знак=старший бит), имеет размер 2 байта (16 двоичных разрядов), и кодируется в дополнительном коде.

Формат long unsigned integer предполагает, что число C – целое без знака (не отрицательное) имеет размер 2 байта (16 двоичных разрядов) и кодируется в прямом коде.

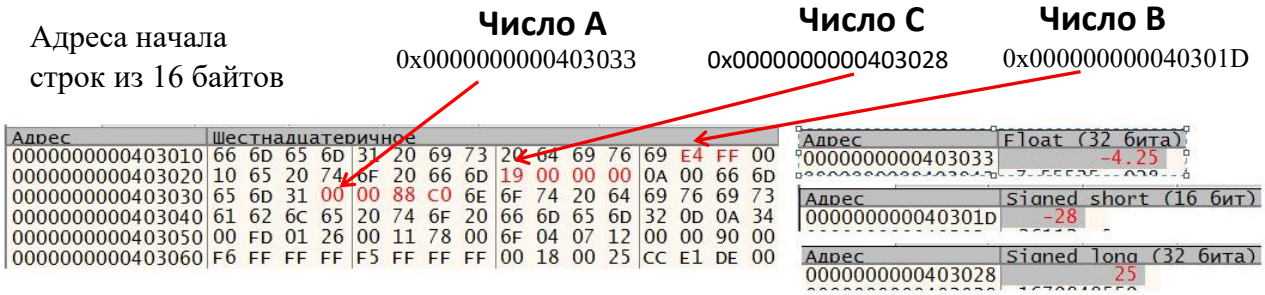


Word Signed Integer или Short Signed Integer,

Doubleword Unsigned Integer или Long Unsigned Integer,

Single Precision Floating Point или Binary32

Найдём эти числа в дампе памяти.



На рисунке в задании представлен шестнадцатеричный код байтов памяти, где пары шестнадцатеричных цифр задают содержимое одного байта. Тогда, согласно заданию, надо сравнить числа со следующими кодами

- двухбайтовый код числа **A** по адресу 0x0000000000403041 – 00 00 88 C0,
- четырёхбайтовый код числа **B** по адресу 0x000000000040301D – E4 FF,
- четырёхбайтовый код числа **C** по адресу x0000000000403028 – 19 00 00 00.

Согласно архитектуре процессоров Intel они используют «Little Endian» принцип размещения многобайтовых величин, что означает расположения числа, начиная с младших его байтов, и адресом числа является адрес его младшего байта. Т.е. правильная привычная нам запись шестнадцатеричных кодов заданных чисел будет следующая:

- A = C0 88 00 00

  или в двоичном представлении 1100 0000 1000 1000 0000 0000 0000 0000,
- B = FF E4

  или в двоичном представлении 1111 1111 1110 0100,
- C = 00 00 00 19

  или в двоичном представлении 0000 0000 0000 0000 0000 0000 0001 1001.

Число A действительное формата ЧПЗ предполагает, что первым слева (старшим) разрядом числа является его знак: («+» кодируется двоичным нулём, «–» кодируется двоичной единицей). Первая двоичная цифра числа «1», значит число A отрицательное. Формат binary32 предполагает следующую структуру разрядов:

Число	Знак числа	Экспонента (смещенный порядок)	Дробная часть мантиссы	Мантисса с целой частью
A	A[31]=1 («–»)	A[30÷23]= 10000001 (129)	A[22÷0]= 000 100 00	1,0001

У числа А истинный порядок вычисляется как разность экспоненты, которая хранится в разрядах 30÷23 и константы +127 и равен 129 – 127 = 2.

Согласно стандарту хранения ЧПЗ в нормализованном формате его целая часть всегда равна единице, которая не хранится в разрядах числа (но подразумевается). Потому мы допишем её к дробной части, которая храниться в разрядах 22÷0.

В результате мы получим следующее число ЧПЗ (в двоичной системе):

$A_2 = -1{,}00010000000000000000 \times 2^2 = -(100{,}01)_2 = -(1\times2^2 + 1\times2^{-2}) = -(4+0{,}25) = -4{,}25.$

Адрес	Float (32 бита)
0000000000403033	-4.25

$A_{10} = -4{,}625$.

Число В представлено дополнительным кодом, который имеет следующую структуру разрядов. Первым слева (старшим) разрядом числа является его знак: («+» кодируется двоичным нулём, «–» кодируется двоичной единицей). Как видим, число В – отрицательное.

15	14 0
Знак	Дополнительный код
1	111 1111 1110 0100

Переведем его в прямой код $[B_2]$пр=0000…0001 1100

и десятичную систему счисления с учетом знака: $B_{10} = -28$.

Число С представлено прямым кодом (без знака, положительное), который имеет следующую структуру разрядов. Первым слева (старшим) разрядом числа является его старший бит в двоичном представлении.

31 0
Прямой код
0000 0000 0000 0000 0000 0000 0001 1001

Переведем его в десятичную систему счисления: $C_{10} = 25$

Сравнивая три числа, получим, что В(–28) < А(–4,25) < С(25).

Ответ: В<А<С или С>А>В

Критерии оценивания решения задачи № 3

Оценивается знание архитектуры компьютера (память и система команд); понятность изложения хода решения, полнота ответов. За полностью верное решение дается 20 баллов.

Что оценивается	Баллы
А) Поиск числа в памяти по указанному адресу	**6**
• *Все числа найдены верно*	6 баллов
• *Есть ошибка в определении адреса одного числа*	–1 балл
• *Есть одна ошибка в определении длины кода числа*	–1 балл
• *Все числа найдены неверно*	0
Б) Запись байтов числа из памяти согласно архитектуре процессора	**3**
• *Числа представлены верно*	3 балла
• *Допущена ошибка при определении порядка байтов в одном числе*	–1 балл
• *Представление чисел не выполнено*	0

В) Представление чисел согласно формату представления разных типов данных	**6**
• *Числа представлены верно*	6 баллов
• *Допущена ошибка при представлении ЧПЗ*	–2 балла
• *Допущена ошибка при представлении одного целого числа*	–2 балла
• *Перевод чисел не выполнен*	0
Г) Вычисление десятичного значения чисел согласно стандарту представления данных	**4**
• *Числа представлены верно*	4
• *Допущена ошибка при вычислении десятичного значения ЧПЗ*	–2 балла
• *Допущена ошибка при вычислении десятичного значения в одном целом числе*	–1 балл
• *Перевод чисел не выполнен*	0
Д) Сравнение чисел согласно правилам машинной арифметики	**1**
• *Сравнение выполнено верно*	1
• *Сравнение выполнено с ошибками или не выполнено*	0

Рекомендуемая литература для подготовки к решению задания № 3

1. IEEE 754 (2008/2019) – IEEE Standard for Floating-Point Arithmetic http://ali.ayad.free.fr/IEEE_2008.pdf
2. Little endian and Big endian Concept with programming Examples. https://aticleworld.com/little-and-big-endian-importance/
3. Intel® 64 and IA-32 Architectures Software Developer's Manual/Volume 1: Basic Architecture (1.3.1 Bit and Byte Order; Chapter 4 Data Types) https://www.intel.com/content/www/us/en/architecture-and-technology/64-ia-32-architectures-software-developer-vol-1-manual.html

ЗАДАЧА № 4. Технологии программирования

Задание

Пусть имеется строка, состоящая из открывающих и закрывающих круглых скобок. Последовательность называется правильной, если число открывающих скобок равно числу закрывающих, и для любой позиции последовательности количество отрывающих скобок слева от нее не больше количества закрывающих. Введем определение глубины скобочной конструкции. Заменим открывающую скобку на 1, а закрывающую на -1, обозначим такое упорядоченное множество (массив) как L. Тогда глубина D(k) скобочной конструкции на позиции k будет равна количеству некомпенсированных открывающихся скобок до k-го символа включительно:

$$\boldsymbol{D(k) = \sum_{i=1}^{k} L[i]}.$$

Пусть N – это количество скобок в последовательности (N – четное число), и есть числа K1 и K2: 1<=K1<=K2<=N/2. Требуется подсчитать количество способов правильной расстановки скобок при условии, что максимальная глубина скобочной конструкции не превышает K2, и хотя бы одна скобочная конструкция имеет глубину не менее K1.

Например, для N=8, K1=2, K2=3 скобочная конструкция "()()()()" не удовлетворяет заданному условию, т.к. максимальная глубина скобок равна 1 и она меньше K1. Последовательность "(()())()" удовлетворяет условию, т.к. максимальная глубина 2 и она не

меньше K1. Вариант "(())()()" также удовлетворяет условию, а последовательность "((()))" не удовлетворяет условию, т.к. максимальная глубина 4 больше, чем K2.

ЗАДАНИЕ: написать функцию, вычисляющую количество корректных расстановок скобок для заданных значений N, K1, K2. Решение должно иметь минимальную временную сложность и требовать минимальный объем памяти (по возможности). Не разрешается использовать библиотеки, автоматизирующие перебор вариантов. Решение принимается на псевдокоде или на языках программирования C, C++, Python, Pascal.

Задача № 4. Решение

Примеры программ:

1. Решение с помощью 1-мерного фазового пространства (ёмкостная сложность O(K2), временная O(N*K2)): 20 баллов.

```
def Count(N, K1, K2):
   if N<2 or N%2!=0 or K1>K2 or K2>N//2: return 0
   def DP(N, K):
      dp=[0]*(K+1)
      dp[0]=1
      for i in range(N):
         dp_new=[0]*(K+1)
         for j in range(K+1):
            if j<K:
               dp_new[j]+=dp[j+1]
            if j>0:
               dp_new[j]+=dp[j-1]
         dp=dp_new
      return(dp[0])
   return DP(N,K2) - DP(N,K1-1)
```

2. Решение с помощью 2-мерного фазового пространства (ёмкостная сложность O(N*K2), временная O(N*K2)): 16 баллов.

```
def Count(N, K1, K2):
   if N<2 or N%2!=0 or K1>K2 or K2>N//2: return 0
   def DP(N, K):
      dp=[[0]*(K+1) for _ in range(N+1)]
      dp[0][0]=1
      for i in range(1,N+1):
         for j in range(K+1):
            if j<K:
               dp[i][j]+=dp[i-1][j+1]
            if j>0:
               dp[i][j]+=dp[i-1][j-1]
      return(dp[N][0])
```

```
    return DP(N,K2) - DP(N,K1-1)
```

3. Рекурсивное решение с ограничением ветвления: 13 баллов.

```
def Count(N,K1,K2, i=0, Skob=0, FK1=False):
    if i+Skob>N: return 0 # Ограничение ветвления
    if i>=N:
        return int(FK1 and Skob==0)
    if Skob==K1:
        FK1=True
    if Skob==0:
        return Count(N, K1,K2,i+1, Skob+1, FK1)
    if Skob<K2:
        return Count(N, K1,K2,i+1, Skob+1, FK1) + Count(N, K1,K2,i+1, Skob-1, FK1)
    return Count(N, K1,K2,i+1, Skob-1, FK1)
```

4. Рекурсивное решение (полный перебор вариантов расстановки скобок до глубины скобочной конструкции K2): 10 баллов.

```
def Count(N,K1,K2, i=0, Skob=0, FK1=False):
    if i>=N:
        return int(FK1 and Skob==0)
    if Skob==K1:
        FK1=True
    if Skob==0:
        return Count(N, K1,K2,i+1, Skob+1, FK1)
    if Skob<K2:
        return Count(N, K1,K2,i+1, Skob+1, FK1) + Count(N, K1,K2,i+1, Skob-1, FK1)
    return Count(N, K1,K2,i+1, Skob-1, FK1)
```

5. Решение с временной сложностью O(2^N)): 8 баллов.

```
def Count(N,K1,K2):
    c=0
    for L in range(2**N):
        Line=bin(L)[2:]
        Line=("0"*(N-len(Line)))+Line
        Skob=0
        FK1=False
        for i in range(N):
            if Line[i]=='0':
                Skob-=1
            else:
                Skob+=1
            if Skob>K2 or Skob<0: break
            if Skob>=K1: FK1=True
        else:
            if FK1 and Skob==0: c+=1
```

```
return c
```

Критерии оценивания решения задачи № 4

Оцениваются умение грамотного составления алгоритма решения вычислительной задачи с наименьшей вычислительной сложностью, оценки вычислительной и емкостной сложностей алгоритма, знание языка программирования высокого уровня; грамотного описания алгоритма. За полностью верное решение дается 20 баллов.

Критерии	Оценка
Основные критерии	
Решение с помощью 1-мерного фазового пространства (ёмкостная сложность алгоритма O(K2), временная O(N*K2))	20 баллов
Решение с помощью 2-мерного фазового пространства (ёмкостная сложность O(N*K2), временная O(N*K2))	16 баллов
Решение с полиномиальной емкостной и временной сложностями	14 баллов
Рекурсивное решение с ограничением ветвления	13 баллов
Рекурсивное решение (полный перебор вариантов расстановки скобок до глубины скобочной конструкции K2)	10 баллов
Решение с временной сложностью O(2^N)	8 баллов
Дополнительные критерии	
Семантические (смысловые) ошибки	-2 балла за каждую
Синтаксические ошибки	-1 балл за каждую
Наличие корректных пояснений к программе	+1 или +2 балла
Отдельные приемы, ускоряющие вычислительный процесс или снижающие необходимый объем памяти для работы программы	+1 балл

Рекомендуемая литература для подготовки к решению задания № 4

1. Яковлев, Игорь Вячеславович. Комбинаторика для олимпиадников / И. В. Яковлев. – 2-е изд., доп. – Москва: Изд-во МЦНМО, 2016. – 77 с.
2. Дольников, Владимир Леонидович. Основные алгоритмы на графах: текст лекций / В. Л. Дольников, О. П. Якимова; М-во образования и науки Российской Федерации, Ярославский гос. ун-т им. П. Г. Демидова. – Ярославль : Ярославский гос. ун-т им. П. Г. Демидова, 2011. – 78 с. (Глава 5)
3. Шень А. Программирование: теоремы и задачи. – 2-е изд., испр. и доп. – М. МЦНМО, 2004. – 296 с. Главы 2,3.
4. Липский, Витольд. Комбинаторика для программистов / В. Липский; Пер. с польского В. А. Евстигнеева, О. А. Логиновой; Под ред. А. П. Ершова. – Москва: Мир, 1988. – 213 с.

ЗАДАЧА №5. Компьютерные сети

Задание

После модернизации сети **с Fast Ethernet до Gigabit Ethernet** организация планировала увеличить пропускную способность сети для внедрения файлового хранилища большой емкости и использования его для хранения файлов Backup.

1. ОС: Windows Server 2025.
2. Размер ежедневного файла архивных копий составляет 200 Гб.
3. Скорость передачи данных канала: 1 Гбит/с
4. Протокол передачи данных: FTP.
5. Настройки сети по умолчанию.
6. RTT = 20 мс.

Задания

1. Рассчитать максимальную скорость загрузки файла архивных копий.
2. Рассчитать время загрузки файла архивных копий.
3. Вычислить эффективность K = T2/T1 от повышения пропускной способности канала передачи данных между сервером и хранилищем (T1 – время загрузки до модернизации, T2 – время загрузки после модернизации).
4. Сделать выводы и дать рекомендации по изменению сложившейся ситуации.

Задача № 5. Решение

Протокол FTP для передачи данных использует протокол TCP. Значит, все расчеты будут касаться пропускной способности протокола TCP.

Следует выделить два наиболее важных фактора для успешной передачи данных по протоколу TCP:

- **размер TCP-окна (буфер)** – количество байт, которое одна из сторон готова принять без подтверждения;
- **Круговая задержка (латентность) передачи-приёма** (англ. round-trip time, RTT) – это время, затраченное на отправку сигнала, плюс время, которое требуется для подтверждения, что сигнал был получен. В контексте компьютерных сетей сигнал – как правило, пакет данных.

Формула расчета пропускной способности TCP:

Для расчета пропускной способности нужно два параметра:

W – размер TCP-окна. Стандартное TCP-окно Windows-машины размером 64 КБ = 65 536 Б = 65 536 * 8 = 524 288 бит

RTT = 20 мс

Пропускная способность TCP (бит/с) = Размер TCP-окна (бит) / Круговая латентность (с)

Пропускная способность TCP (бит/с) = W(бит) / RTT (с).

Протокол прикладного уровня FTP используется для передачи данных протокол TCP/

Рассчитаем скорость передачи по протоколу FTP Ethernet-канала между серверами с круговой задержкой 20 мс. Необходимо отправить большой файл с одного сервера на другой сервер по протоколу FTP. На какую реальную пропускную способность можно рассчитывать?

Сначала необходимо перевести размер TCP-окна из байтов в биты. В данном случае будет задействовано стандартное TCP-окно Windows-машины размером 64 КБ = 65 536 Б = 65 536 * 8 = 524 288 бит.

Затем необходимо взять размер TCP-окна в битах и разделить его на круговую латентность канала, выраженную в секундах. Для целей данных расчетов 20 мкс превращается в 0.020 с.

1. Рассчитать максимальную скорость загрузки файла архивных копий.

   Для расчета пропускной способности нужно два параметра:
   W – размер TCP-окна. Стандартное TCP-окно Windows-машины размером 64 КБ = 65 536 Б = 65 536 * 8 = 524 288 бит
   RTT = 20 мс
   Пропускная способность TCP (бит/с) = Размер TCP-окна (бит) / Круговая латентность (с)
   Пропускная способность TCP (бит/с) = W(бит) / RTT (с).
   Протокол прикладного уровня FTP используется для передачи данных протокол TCP/
   Рассчитаем скорость передачи по протоколу FTP Ethernet-канала между серверами с круговой задержкой 20 мс. Необходимо отправить большой файл с одного сервера на другой сервер по протоколу FTP. На какую реальную пропускную способность можно рассчитывать?
   Сначала необходимо перевести размер TCP-окна из байтов в биты. В данном случае будет задействовано стандартное TCP-окно Windows-машины размером 64 КБ = 65 536 Б = 65 536 * 8 = 524 288 бит.
   Затем необходимо взять размер TCP-окна в битах и разделить его на круговую латентность канала, выраженную в секундах. Для целей данных расчетов 20 мкс превращается в 0.020 с.
   Максимальная скорость загрузки для при использовании протокола TCP = 524 288 бит / 0.020 с = 26 214 400 бит/с = 26.2 Мбит/с.
   И, очевидно, не зависит от скорости канала передачи данных, а только от размера от размера окна TCP и задержки RTT (круговой латентности).

2. Рассчитать время загрузки файла архивных копий.

   200 Гб = 1 717 986 918 400 Бит (бит)
   T1 = 1 717 986 918 400 / 26 214 400 = 65 536 (с) = 65 536 / 3 600 = 18,2 (ч)
   Исходя из расчетов в п.1, время загрузки по каналу передачи 1 Гбит/с останется тем же самым.
   T2 = 1 717 986 918 400 / 26 214 400 = 65 536 (с) = 65 536 / 3 600 = 18,2 (ч)
3. Вычислить эффективность K = T2/T1 от повышения пропускной способности канала передачи данных между сервером и хранилищем (T1 – время загрузки до модернизации, T2 – время загрузки после модернизации).

   Исходя из расчетов в п. 2

   K = T2/T1 = 18,2/18,2 = 1.

4. Сделать выводы и дать рекомендации по изменению сложившейся ситуации.

Выводы:

Повышение скорости канала между серверами с прежней задержкой RTT = 20 мс не повлияет на скорость передачи по протоколу TCP и время загрузки.

Рекомендации:

Можно не увеличивать TCP-окно и не устанавливать WAN-ускорители. Необходимо разбить файл на несколько частей и использовать многопоточность, тем самым можно использовать канал на его максимальной пропускной способности.

Критерии оценивания решения задачи № 5

Оценивается знание модернизация компьютерной сети; понятность изложения хода решения, полнота ответов. За полностью верное решение дается 20 баллов.

Что оценивается	Баллы
1.Рассчитать максимальную скорость загрузки файла архивных копий	**6**
• *Расчеты выполнены верно*	6 баллов
• *Есть одна ошибка в выборе параметров для расчетов*	–1 балл
• *Есть одна ошибка в рассуждениях*	–1 балл
• *Расчеты выполнены неверно*	0
2.Рассчитать время загрузки файла архивных копий	**5**
• *Расчеты выполнены верно*	5 баллов
• *Есть одна ошибка в выборе параметров для расчетов*	–1 балл
• *Есть одна ошибка в рассуждениях*	–1 балл
• *Расчеты выполнены неверно*	0
3.Вычисление эффективности K = T2/T1	**5**
• *Расчеты выполнены верно*	5 баллов
• *Одна ошибка в предварительных расчетах*	–1 балл
• *Есть одна ошибка в выборе параметров для расчетов*	–1 балл
• *Расчеты выполнены неверно*	0
4.Сделать выводы и дать рекомендации	**4**
• *Сделаны правильные выводы и даны выполнимые рекомендации*	4 балла
• *Сделаны неправильные выводы*	- 1 балл
• *Даны невыполнимые рекомендации*	- 1 балл
• *Сделаны неправильные выводы и даны невыполнимые рекомендации*	0

Рекомендуемая литература для подготовки к решению задания № 5

1. Таненбаум Э., Фимстер Н., Уэзеролл Д. Компьютерные сети. 6-е изд. – СПб.: Питер, 2023. – 992 с.: ил. (Серия «Классика computer science»). ISBN 978-5-4461-1766-6
2. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы (учебник для ВУЗ’ов). – СПб.: Питер, 2017. – 992 с.
3. Борисов С.П. Сети и телекоммуникации [Электронный ресурс] Учебное пособие / Борисов С.П. – М.: МИРЭА – Российский технологический университет, 2018.