

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ  
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО  
ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»  
Физико-технический факультет**

**УТВЕРЖДЕНО:**

на заседании Ученого совета  
физико-технического факультета  
протокол № 6 от 15.02.2019 г.

председатель совета

\_\_\_\_\_ Фоменко С.А.



**ПРОГРАММА**

профильного экзамена по специальности  
для абитуриентов, поступающих на обучение  
по образовательной программе

**МАГИСТРАТУРЫ**

на направление подготовки  
09.04.01 Информатика и вычислительная техника

**Донецк, 2019**

## **Содержание программы**

1. Введение	3
2. Общие положения	3
3. Перечень вопросов для подготовки к сдаче вступительного испытания	4
4. Структура билета	11
5. Образец бланка ответов	17
6. Критерии оценивания письменных ответов на вступительном испытании	18
7. Список рекомендованной литературы	19

## **1. Введение**

Целью вступительного испытания в Магистратуру по направлению подготовки 09.04.01 «Информатика и вычислительная техника» является проверка уровня и качества общей, общепрофессиональной и специальной подготовки абитуриентов по циклам дисциплин, предусмотренных Общей образовательной программой по группе направлений подготовки 09.00.00 «Информатика и вычислительная техника».

## **2. Общие положения**

Испытание проводится в письменной форме и включает в себя вопросы по дисциплинам «Базы данных», «Архитектура ЭВМ и микроконтроллеров», «Операционные системы», «Дискретная математика», «Основы программирования», «Программирование», «Сети и телекоммуникации» знания и навыки в области, которых, позволяют успешно выполнить задания по вступительному испытанию.

В программе используется материал теоретического и прикладного характера в виде тестов и одного открытого практического задания.

Порядок проведения вступительного испытания определяется Положением о приемной комиссии ГОУ ВПО «Донецкий национальный университет».

### **3. Перечень вопросов для подготовки к сдаче вступительного испытания и список рекомендованной литературы**

#### **1. Базы данных**

- 1.1. Архитектура баз данных
- 1.2. Логические и физические модели данных
- 1.3. Основные понятия и термины реляционной базы данных
- 1.4. Проектирование баз данных
- 1.5. Реляционная алгебра. Реляционное исчисление (на кортежах и на доменах).
- 1.6. Операторы SQL, их определение
- 1.7. Определение и содержание понятий "транзакция", "журнализация изменений в БД"
- 1.8. Нормализация отношений. Определение нормальных форм.
- 1.9. Определение целостности БД.
- 1.10. Функциональные и многозначные зависимости в отношениях.
- 1.11. Расподилени БД. Однородные, неоднородно, гетерогенные СУБД.
- 1.12. Принципы и понятия реализации клиент-серверной технологии.

#### **2. Архитектура ЭВМ и микроконтроллеров**

- 2.1. Понятие конфигурации ЭВМ, архитектуры ЭВМ.
- 2.2. Понятие архитектуры процессора.
- 2.3. Структура 16-битного процессора Intel 8088. Главные регистры.
- 2.4. Структура 16-битного процессора Intel 8088. Регистр флагов.
- 2.5. Алгоритм работы процессора Intel 8088.
- 2.6. Процессор Intel 8088. Язык Ассемблер. Типы данных. Псевдооператор определения данных.
- 2.7. Процессор Intel 8088. Организация памяти. Сплошная и сегментная модели. Псевдооператор определения сегментов и процедур.
- 2.8. Процессор Intel 8088. Режимы адресации к памяти.
- 2.9. Система машинных команд процессора Intel 8088. Типы команд. Команды пересылки данных.
- 2.10. Система машинных команд процессора Intel 8088. Команды арифметики.
- 2.11. Система машинных команд процессора Intel 8088. Команды передачи управления.
- 2.12. Система машинных команд процессора Intel 8088. Логические команды.
- 2.13. Система машинных команд процессора Intel 8088. Команды манипулирования битами.
- 2.14. Система машинных команд процессора Intel 8088. Команды обработки строк.
- 2.15. Система машинных команд процессора Intel 8088. Команды прерываний.
- 2.16. Система машинных команд процессора Intel 8088. Команды управления работой процессора.

- 2.17. Язык Ассемблер процессора Intel 8088. Макроопределение.
- 2.18. Арифметический сопроцессор 8087. Типы данных.
- 2.19. Арифметический сопроцессор 8087. Главные регистры.
- 2.20. Арифметический сопроцессор 8087. Система кома

### **3. Операционные системы**

- 3.1. Функции, выполняемые операционной системой.
- 3.2. Классификация ОС по особенностям алгоритмов управления процессором.
- 3.3. Назначение мобильных операционных систем.
- 3.4. Системы пакетной обработки. Критерии эффективности.
- 3.5. Операционные системы, построенные на базе микроядерного подхода, и системы с монолитным ядром. Эффективность функционирования.
- 3.6. Концепция множественности прикладных сред.
- 3.7. Компоненты сетевой операционной системы.
- 3.8. Редиректор. Выполняемые функции.
- 3.9. Функции подсистемы управления процессами.
- 3.10. Состояния процесса.
- 3.11. Дескриптор процесса.
- 3.12. Очереди процессов. Структуры для описания очередей.
- 3.13. Алгоритмы планирования процессов, основанные на квантовании.
- 3.14. Критические секции. Способы обеспечения взаимного исключения при реализации критических секций.
- 3.15. Алгоритм Дейкстры. Определение примитивов P и V.
- 3.16. Монитор - средство синхронизации процессов. Вызов процессами процедур монитора.
- 3.17. Нити. Общие ресурсы.
- 3.18. Нити. Собственные ресурсы.
- 3.19. Функции ОС по управлению памятью.
- 3.20. Виртуальное адресное пространство процесса.
- 3.21. Загрузчик, что перемещает. Преобразование виртуального адреса в физический.
- 3.22. Методы распределения памяти без использования дискового пространства.
- 3.23. Метод распределения памяти фиксированными разделами. Недостаток метода.
- 3.24. Задача подсистемы управления памятью при реализации метода распределения памяти размерами переменной величины.
- 3.25. Задачи, решаемые виртуальной памятью.
- 3.26. Страничное распределение памяти. Описание виртуального адресного пространства процесса.
- 3.27. Страничное распределение памяти. Время преобразования виртуального адреса в физический.
- 3.28. Достоинства сегментного распределения памяти.
- 3.29. Странично-сегментное распределение памяти.

- 3.30. Свопинг.
- 3.31. Кэш-память. Определение.
- 3.32. Структура кэш-памяти.
- 3.33. Адресация в системе с кэш-памятью.
- 3.34. Среднее время доступа к данным в системе с кэш-памятью.
- 3.35. Функции операционной системы при управления устройствами ввода-вывода.
- 3.36. Механизм выполнения ввода - вывода операционной системой.
- 3.37. Слои программного обеспечения ввода - вывода операционной системы.
- 3.38. Функции независимого от устройств слоя программного обеспечения ввода-вывода.
- 3.39. Спулинг.
- 3.40. Структура записи каталога операционной системы UNIX.
- 3.41. Логическая структура файла в операционной системе UNIX.
- 3.42. Функции базового уровня файловой системы.
- 3.43. Проблемы при отображении файлов в адресное пространство выполняемого процесса.
- 3.44. Добавление драйвера к цепочке вызова некоторого оборудования в современной многоуровневой архитектуре файловых систем.

#### **4. Дискретная математика**

- 4.1. Операции над множествами и их свойства.
- 4.2. Понятие отношения. Свойства и виды бинарных отношений.
- 4.3. Понятие функции. Сюръекция, инъекция, биекция.
- 4.4. Основные понятия теории графов. Деревья. Вершинные и реберные покрытия.
- 4.5. Правильное раскраска графа.
- 4.6. Элементы комбинаторики: комбинации, перестановки, размещения. Свойства биномиальных коэффициентов.
- 4.7. Булевы функции. ДНФ, КНФ, поленом Жегалкина.
- 4.8. Функциональная полнота систем булевых функций. Теорема Поста.
- 4.9. Минимизация булевых функций.
- 4.10. Конечные автоматы. Способы задания автоматов.

#### **5. Основы программирования**

- 5.1. Алгоритм. Свойства алгоритма. Виды представления алгоритма.
- 5.2. Основные парадигмы и подходы современного программирования.
- 5.3. Интегрированная среда разработки MS Visual Studio. Работа с проектами. Отладка программ. Конфигурации Debug и Release.
- 5.4. Основные элементы фреймворка .NET: общезыковая среда выполнения (CLR) и базовая библиотека классов (BCL).
- 5.5. Процесс создания программы. Компиляция и линковка. Пункт меню Build и Debug в MS Visual Studio.
- 5.6. Простые типы данных .NET. Преобразование и приведение простых типов.

Типизация и ключевое слово var.

5.7. Арифметические операции. Постфиксный и префиксный инкремент (декремент). Математические операции (возведение в степень, тригонометрия, логарифмы).

5.8. Простые и составные логические выражения. Логические операторы. Особенности "укороченных" логических операторов. Таблицы истинности.

5.9. Ветвления. Операторы if, switch, goto. Тернарный оператор (?).

5.10. Циклы. Операторы for, while, do while, continue, break.

5.11. Массивы. Оператор foreach. Виды двумерных массивов. Инициализация одномерных и двумерных массивов. Генерация случайных чисел.

5.12. Пользовательские функции. Ключевые слова params, ref, out. Параметры по умолчанию. Оператор return.

5.13. Рекурсия. Базовый случай, шаг рекурсии. Стек вызовов. Рекурсивные алгоритмы нахождения НОД двух чисел и факториала числа.

5.14. Работа со строками как с массивом символов. Тип string. Основные операции со строками (поиск, вставка, удаление, замена подстроки).

5.15. Регулярные выражения. Основные синтаксические элементы регулярных выражений. Квантификаторы.

5.16. Тип данных - перечисление. Синтаксис. Занимаемый объем памяти. Модификации. Примеры.

5.17. Структуры в C#. Работа с датой и временем в C#.

5.18. Структура программы. Переменные и константы. Идентификаторы. Операторы. Выражения.

5.19. Абстрактные синтаксические деревья. РБНФ-выражения. Примеры.

## **6. Программирование**

6.1. Алфавит и лексемы языка C ++. Идентификаторы, разделители, операции, комментарии языка. Базовые типы данных. Константы базовых типов. Состав выражение на C ++. Особенность выражение в C ++.

6.2. Приоритетность операций и порядок обработки операндов в C ++. Правила преобразований значений операндов из одного типа в другой для операций в C ++ и порядок автоматического приведения типов в выражении на C ++. Операция преобразования типа в C ++.

6.3. Управляющие операторы языка C ++. Организации разветвления и цикла. Операторы, расширяющие возможности передачи управления.

6.4. Особенности языка C ++. Структура программы на C ++. Команды препроцессору. Организация функций в C ++.

6.5. Область действия переменной в C ++. Классы памяти в C ++. Организация ввода - вывода в C и C ++. Библиотеки stdio.h и iostream.h.

6.6. Свойства консольного приложения в среде Visual C ++ 6.0. Состав проекта консольного приложения. Этапы разработки.

- 6.7. Понятие указателя в C++. Типы указателей. Операции над указателями. Понятие ссылки. Различия в объявлении и использовании ссылки и указателя. Работа с динамической областью памяти.
- 6.8. Организация структурированных типов в языке C++. Строки, массивы, перечисления, структуры, объединения, поля бит. Способ хранения, виды доступа к элементам. Организации хранения и обработки в статической и динамической памяти.
- 6.9. Понятие инкапсуляции, класса и объекта. Принципы отделения интерфейса от реализации. Спецификаторы доступа public, private, protected.
- 6.10. Особенности объявления и использования функций - членов (методов) класса. Доступ к методам; функции-члены, встраиваемые. Перегруженные функции - члены. Переопределения и сокрытия методов класса. Оператор разрешения области видимости.
- 6.11. Особенности объявление и использование данных - членов класса. Доступ к данным. Оператор разрешения области видимости. Использование конструкторов для инициализации данных класса. Явные и неявные конструкторы. Конструкторы по умолчанию. Перегруженные конструкторы; конструкторы, копируют. Конструкторы конструкторов.
- 6.12. Особенности объявления и использование деструкторов класса. Явные и неявные деструкторы. Преимущества использования виртуальных деструкторов. Использование деструкторов при работе с массивом объектов класса.
- 6.13. Понятие наследования. Схемы взаимодействия объектов. Формальный запись определения наследования. Типы наследования. Виды наследования.
- 6.14. Понятие полиморфизма. Формы проявления полиморфизма. Назначение и особенности виртуальных функций. "Раньше" и "позже" связки. Преимущества динамического связывания. Абстрактные классы.
- 6.15. Понятие шаблона. Шаблонная функция и шаблонный класс. Особенности компиляции шаблонных функций и классов. Шаблоны и статические элементы.
- 6.16. Организация дружественных функций и классов. Особенности дружественности для шаблонов. Виды дружественности для функций и для классов.
- 6.17. Особенности реализации учетных структур на C++. Организация однонаправленный и двунаправленного списков и базовые операции с такими списками.
- 6.18. Стандартная библиотека C++. Состав библиотеки и краткая характеристика разделов библиотеки.
- 6.19. Основные термины библиотеки STL. Понятие контейнера и схема организации контейнера. Иерархия контейнерных классов STL. Понятие итераторов и иерархия итераторов. Понятие алгоритма, аллокатора, адаптера.
- 6.20. Последовательные контейнеры. Организация работы с вектором, списком и противнем средствами STL.



- 6.21. Ассоциативные контейнеры. Организация работы с отражениями (деревьями, множествами) средствами STL.
- 6.22. Группы алгоритмов в STL. Не мутационные алгоритмы и их группы. Алгоритм поиска find (). Мутационные алгоритмы и их группы. Алгоритм слияния merge ().
- 6.23. Классы исключительных ситуаций. Цель классов исключительных ситуаций и иерархия стандартных исключительных ситуаций. Обработка логических ошибок и ошибок выполнения.
- 6.24. Работа с файлами в C++. Свойства обработки элементов памяти-файла, каталога, диска.

## **7. Сети и телекоммуникации**

- 7.1. Локальные и глобальные вычислительные сети. Топологии сетей. Широкополосная и узкополосная передача. Полудуплексная и полнодуплексная передача. Сегменты и магистрали. Одноранговые и клиент - серверные сети.
- 7.2. Эталонная модель OSI. Инкапсуляция данных. Физический уровень. Канальный уровень. Формат кадра. Управление доступом к среде. Спецификации физического уровня. Сетевой уровень. Адресация. Фрагментация, Маршрутизация.
- 7.3. Сетевое оборудование. Сетевые подключения. Мосты, коммутаторы, маршрутизаторы и шлюзы.
- 7.4. Сетевое программное обеспечение. Сетевые операционные системы. Сетевые клиенты. Служба каталога.
- 7.5. Канальный уровень. Технология CSMA / CD, Сети Ethernet. Локальные сети на разделяемой среде, Ethernet на коаксиальном кабеле, Ethernet на витой паре, Сети Token Ring и FDDI. Преимущества и недостатки разделяемой среды
- 7.6. Транспортный уровень
- 7.7. Сеансовый уровень
- 7.8. Представительный уровень
- 7.9. Прикладной уровень
- 7.10. Стандартные стеки протоколов
- 7.11. Простейший случай локальной компьютерной сети. Совместное использование ресурсов.  
Доступ к периферийному устройству через сеть Передача данных по линиям связи.
- 7.12. Мережевое программное обеспечение. Сетевые службы и сервисы. Сетевая операционная система. Сетевые программы.
- 7.13. Основы адресации, типы адресов и преобразования адресов.
- 7.14. Классовая модель IP-адресации, классы IP-сетей
- 7.15. Бесклассовая модель IP-адресации, маска переменной длины. Назначение IP-адресов
- 7.16. Виртуальные локальные сети. пользовательские фильтры
- 7.17. Логическое разделение сети на виртуальные локальные сети

- 7.18. Коммутируемые сети Ethernet. Принцип работы коммутаторов Ethernet. Протокол STP.
- 7.19. Беспроводные локальные сети. стандарты IEEE 802.11
- 7.20. Протокол межсетевого взаимодействия. Формат IP-пакета.
- 7.21. Таблица маршрутизации. маршрутизации без масок.
- 7.22. Взаимодействие протоколов IP. ARP. Ethernet и DNS.
- 7.23. Структуризация сетей на основе масок. Маршрутизация с масками.
- 7.24. CIDR и маршрутизация.
- 7.25. Фрагментация IP-пакетов
- 7.26. Протоколы транспортного уровня TCP и UDP. Порты и сокет
- 7.27. Протокол UDP
- 7.28. Протокол TCP и TCP-сегменты. Логические соединения - основа надежности TCP.
- 7.29. Протоколы маршрутизации. Общие свойства протоколов маршрутизации
- 7.30. Протокол OSPF. Взаимодействие протоколов маршрутизации
- 7.31. Вспомогательные протоколы и средства стека TCP / IP.
- 7.32 Протокол ICMP. Утилиты traceroute. ping. Netstat.

## 4. Структура билета

### ПАКЕТ 1

#### УТВЕРЖДЕНО:

на заседании Ученого совета  
физико-технического факультета  
протокол № 6 от 15.02.2019

### ГОУ ВПО «Донецкий национальный университет» Физико-технический факультет

Профильный экзамен по	<u>СПЕЦИАЛЬНОСТИ</u>
ОП	Магистратура
Форма обучения	Очная, заочная
Направление подготовки:	09.04.01 Информатика и вычислительная техника

Тестовое теоретическое задание

#### ЗАДАНИЕ №1

**Базы данных. Отношение R находится в 3НФ в случае:**

- 1) если все неключевые атрибуты R взаимно зависимы и полностью зависят от первичного ключа;
- 2) если все неключевые атрибуты R взаимно независимы и полностью зависят от первичного ключа;
- 3) если все ключевые атрибуты R взаимно независимы и полностью зависят от первичного ключа;
- 4) если все ключевые атрибуты R взаимно зависимы и полностью зависят от первичного ключа;
- 5) если все атрибуты R взаимно зависимы и полностью зависят от первичного ключа.

#### ЗАДАНИЕ №2

**Архитектура ЭВМ и микроконтроллеров. С помощью какой директивы на языке АССЕМБЛЕРА определяются четыре данных символьного типа?**

- 1) POLE DB '7', '5', '3', '9'
- 2) POLE DQ 1001b
- 3) POLE DW 0CBC3H
- 4) POLE DD 7,5,3,9
- 5) POLE DB 0BH, 0CH, 0DH, 0FH

#### ЗАДАНИЕ №3

**Программирование. Операционная система выполняет следующие две, по сути мало связаны функции:**

- 1) обеспечение программисту удобств посредством предоставления для него расширенной машины и повышение эффективности использования компьютера путем рационального управления его ресурсами;

- 2) предоставление пользователю возможности машинного языка для управления внешними устройствами и распределение внешних устройств между конкурирующими процессами;
- 3) обеспечение программисту удобств посредством предоставления для него расширенной машины и распределение внешних устройств между конкурирующими процессами;
- 4) обеспечение программисту удобств посредством предоставления для него расширенной машины и предоставления пользователю возможностей машинного языка для управления внешними устройствами;
- 5) предоставление пользователю возможности машинного языка для управления внешними устройствами и повышения эффективности использования компьютера путем рационального управления его ресурсами.

#### ЗАДАНИЕ №4

**Дискретная математика. Булевой функцией называется:**

- 1) взаимно однозначное соответствие между булевыми векторами и множеством натуральных чисел;
- 2) функция, определенная на  $n$ -мерном булевом кубе  $B^n$ ,  $n > 0$ , и принимающая значения из множества  $\{0,1\}$ ;
- 3) функция, определенная на множестве  $B^n$  и принимающая значения из множества натуральных чисел;
- 4) функция, определенная на множестве  $B^n$ ;
- 5) среди перечисленных вариантов нет правильного ответа.

#### ЗАДАНИЕ №5

**Сети и телекоммуникации. В качестве какого устройства работает Windows-компьютер, предоставляющий общий доступ в Интернет компьютерам локальной сети?**

- 1) концентратор.
- 2) Маршрутизатора.
- 3) Моста.
- 4) коммутатора.

#### ЗАДАНИЕ №6

**Основы программирования. Что выведет на экран следующая программа?**

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {int k=3,m=5; cout<<25/m <<" and "<<17/k; return 0;}
```

- 1) 0 and 2
- 2) 5 and 5.66667
- 3) 5.0 and 5.66667
- 4) 5 and 5
- 5) Будет выведена ошибка периода выполнения

#### ЗАДАНИЕ №7

**Программирование. Задача редактора связей состоит в следующем:**

- 1) Генерация промежуточного кода (между языком исходным и машинным)

- 2) Выполнение процесса сборки системных программ и связывание их с программой пользователя
- 3) Выполнение процесса сборки программы с предварительно откомпилированными пользовательскими библиотечными программами
- 4) Варианты 2 и 3
- 5) Интерпретация программы пользователя с помощью другой программы

#### ЗАДАНИЕ №8

**Архитектура ЭВМ и микроконтроллеров. Какой шестнадцатилетний код содержится в регистре AL после выполнения фрагмента программы на языке АССЕМБЛЕРА при CF = 0?**

```
MOV CL, 2
MOV AL, 0FH
SAL AL, CL
```

- 1) 3C;
- 2) 0F;
- 3) 1E;
- 4) F0;
- 5) 78.

#### ЗАДАНИЕ №9

**Программирование. При подкачки между памятью и диском:**

- 1) перемещаются страницы процесса, которые давно не использовались;
- 2) перемещается целиком весь процесс;
- 3) перемещаются сегменты процесса, которые давно не использовались;
- 4) перемещаются страницы, которые давно не использовались, и сегменты процесса;
- 5) перемещаются только процессы, выполняющие расчетные задания.

#### ЗАДАНИЕ №10

**Базы данных. Особенности операции сложения нового кортежа в отношении:**

- 1) увеличивает размерность отношения
- 2) меняет схему отношения
- 3) изменяет схему и информационное наполнение отношения
- 4) не вносит изменений в информационное наполнение отношения
- 5) увеличивает объем информационного наполнения отношения

#### ЗАДАНИЕ №11

**Сети и телекоммуникации. Как называется беспроводная ЛВС, в которой не используются точки доступа?**

- 1). Каждый с каждым
- 2). распределенная
- 3). инфраструктура
- 4). Базовая

#### ЗАДАНИЕ №12

**Основы программирования. Как работает следующий оператор:**

```
switch (next_char){ case 'a':case 'A': case 'e':case 'E': case 'i':case 'I': case 'o':case 'O':
case 'u':case 'U': ++vowel_cnt; // «Некоторые действия»}
```

- 1) Если переменная `next_char` равна одному из символов 'a', 'A', 'e', 'E', 'i', 'I', 'o', 'O', 'u', 'U', то переменная `vowel_cnt` увеличивается на единицу и происходит выход из `switch`
- 2) Оператор не работает, так как после `case` должен быть хотя бы один оператор
- 3) Если переменная `next_char` равна одному из символов 'a', 'A', 'e', 'E', 'i', 'I', 'o', 'O', 'u', 'U', то переменная `vowel_cnt` увеличивается на единицу и происходит выход из `switch`, если нет – то выполняются «Некоторые действия»
- 4) Переменная `vowel_cnt` увеличивается на единицу независимо от значения переменной `next_char` и происходит выход из `switch`
- 5) Если переменная `next_char` равна одному из символов 'a', 'e', 'i', 'o', 'u', то переменная `vowel_cnt` увеличивается на единицу, выполняются «Некоторые действия» и происходит выход из `switch`

#### ЗАДАНИЕ №13

**Программирование.** Даны целочисленные переменные `A`, `B` и `C`. Присвоить переменной `A` значение, равное полусумме значений переменных `B` и `C`, причем значения `B` и `C` - четные. Каким из следующих операторов присваивания это можно выполнить?

- 1) `A:=B+C div 2;`
- 2) `A:=(B+C)/2;`
- 3) `A:=0.5*TRUNC(B+C);`
- 4) `A:=B/2 + C/2;`
- 5) `A:=ROUND(0.5*(B+C));`

#### ЗАДАНИЕ №14

**Дискретная математика.** Сколькими способами число  $7^n$  можно представить в виде трех сомножителей отличных от 1? (Представления, различающиеся лишь порядком сомножителей, считаются разными.)

- 1)  $C_{n+4}^n$ ;
- 2)  $C_n^3$ ;
- 3)  $A_n^3$ ;
- 4)  $C_{n-1}^2$ ;
- 5)  $n!$ .

#### ЗАДАНИЕ №15

**Системное программирование и операционные системы.** Структура записи каталога операционной системы UNIX:

- 1) имя файла, расширение, атрибуты, время создания, дата создания, номер первого блока файла, размер файла;
- 2) имя файла, расширение, атрибуты, время создания, дата создания файла;
- 3) № индексного дескриптору, имя файла, расширение, атрибуты файла;
- 4) № индексного дескриптору, имя файла;
- 5) № индексного дескриптору, атрибуты, время создания, дата создания, номер первого блока файла.

#### ЗАДАНИЕ №16

**Базы данных.** Предикаты раздела `WHERE` оператора `SELECT`:

- 1) `ALL`, `DISTINCT`:

- 2) ORDER BY, GROUP BY;
- 3) BETWEEN, IS [NOT] NULL, IN, LIKE;
- 4) HAVING, COUNT;
- 5) ASC, DESC.

#### ЗАДАНИЕ №17

**Сети и телекоммуникации. Какой компонент определяет количество бит IP-адреса, использующихся для представления идентификатора (ID) узла?**

1. основной шлюз
2. номер узла
3. номер сети
4. маска подсети

#### ЗАДАНИЕ №18

**Программирование. Какая из приведенных ниже записей (Borland Pascal 7.0) может являться правильным арифметическим выражением?**

- 1)  $A[2] + 3 * ABS(2 + A[2,3])$
- 2)  $A[COS(X)][2]$
- 3)  $(MAXINT + 1) / 2$
- 4)  $(A + B / (C + (D + 2)))$
- 5)  $SQRT(-SQR(X))$

#### ЗАДАНИЕ №19

**Дискретная математика. Каким классам Поста принадлежит функция  $x \downarrow y$  ?**

- 1) всем;
- 2) ни одному;
- 3) принадлежит  $T_0$  и  $T_1$  и не принадлежит  $S, L, M$ ;
- 4) принадлежит  $S, L, M$  и не принадлежит  $T_0$  и  $T_1$ ;
- 5) принадлежит  $S$  и  $L$  и не принадлежит  $T_0, T_1$  и  $M$ .

#### ЗАДАНИЕ №20

**Архитектура ЭВМ и микроконтроллеров. Какой шестнадцатилетний код содержится в регистре AL после выполнения фрагмента программы на языке АССЕМБЛЕРА при CF = 1?**

```
MOV CL, 3
MOV AL, 0FH
RCR AL, CL
```

- 1) 5C;
- 2) 87;
- 3) 18;
- 4) F1;
- 5) E1;

#### ОТКРЫТЫЙ ВОПРОС

**Базы данных.**

Рассматривается предметная область «Научные разработки». Имеем отношения:  
DEVELOPERS (ID\_dev, name\_dev, city, bonus);

где: ID\_dev - идентификатор разработчика научной продукции,  
name\_dev - имя разработчика научной продукции,  
city - город проживания разработчика,  
bonus - вознаграждение за научную разработку.

**Задача:** Используя SQL-выражение, получить информацию о всех разработчиков (name\_dev, city) в городе Kyiv, которые получили вознаграждение более 0.15.

Председатель приёмной комиссии

С.В. Беспалова

Председатель экзаменационной комиссии

С.А. Фоменко



## 5. Образец бланка ответов

ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»

**ШИФР**

Физико-технический факультет

Профильный экзамен по Специальности

**Оценка**

Лист ответов

Билет № \_\_\_\_\_

### Тестовые задания

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Итого:
1																					
2																					
3																					
4																					
5																					

### Открытый вопрос

## 6. Критерии оценивания письменных ответов на вступительном испытании

Главной целью вступительного испытания является проверка остаточных знаний, полученных в бакалавриате по блоку направлений «Информатика и вычислительная техника».

Вступительные испытания проводятся в письменной форме, включают в себя тестовые задания, а также задания по теоретической и практической части изученных курсов.

Испытание проводится письменно в виде тестов и практического задания; каждый тест оценивается в 4 балла (всего 20 вопросов), открытый практический вопрос оценивается в 20 баллов (всего 1 вопрос). Максимальное количество баллов, полученных за решение всех заданий, составляет **100** баллов.

Соответственно, по ниже приведенной схеме осуществляется перевод баллов в пятибалльную систему:

<b>100-балльная шкала</b>	<b>5-балльная шкала</b>
0-59	«2» (неудовлетворительно)
60-74	«3» (удовлетворительно)
75-89	«4» (хорошо)
90-100	«5» (отлично)

**Примечание.** Лица, получившие 0–59 балла (по 100-балльной шкале) считаются получившими оценку «неудовлетворительно».

Все ответы должны вноситься в листы заданий путем вписывания необходимого ответа. Он заполняется ручкой синего или черного цвета. Обязательно фиксируется номер варианта на листе письменной работы. Никакие лишние пометки на листе письменной работы не допускаются.

Тестовые задания должны быть выполнены в течение 90 минут.

Критерии утверждены ученым советом физико-технического факультета, протокол № 6 от 15.02.2019 года.

## **7. Список рекомендованной литературы**

### **1. Базы данных**

1. К.Дж.Дейт. Введение в системы баз данных. - Диалектика, 2002.
2. Т. Конноли. Базы данных: проектирование, реализация - ВЛМ, 2000.
3. Берко А.Ю. Верес А.Н. Организация баз данных: практический курс. учеб. пособие. - Львовская политехника, 2003.

### **2. Архитектура ЭВМ и микроконтроллеров**

1. Абель Питер. Язык ассемблера для IBM PC и программирование. -М .: Высшая школа.-1992г.
2. Скэнлон Л. Персональные ЭВМ IBM PC и XT. Программирование на языке Ассемблера. -М .: Радио и связь.-1991г.
3. Джордейн Р. Справочник программиста персональных компьютеров типа IBM PC, XT и AT. -М .: Финансы и статистика. -1992г.

### **3. Операционные системы**

1. Бек Л. Введение в системное программирование. -М .: Мир, 1988 г.
2. Джордейн Р. Справочник программиста персональных компьютеров типа IBM PC, XT и AT. -М .: Финансы и статистика. -1992г.
3. Абель Питер. Язык ассемблера для IBM PC и программирования. -М .: Высшая школа. -1992г.

### **4. Дискретная математика**

1. Андерсон Д.А. Дискретная математика и комбинаторики. : Пер. с англ. - М .: Издательский дом «Вильямс», 2004. - 960 с.
2. Капитонова Ю.В., Кривой С.Л., Летичевский А.А., Луцкий Г.М., Печорин М.К. Основы дискретной математики - М .: Наукова думка, 2002. - 581 с.
3. Судоплатов С.В., Овчинникова Е.В. Дискретная математика. - Инфра-М НГТУ, 2009. - 256 с.
4. Шоломова Л.А. Основы теории дискретных логических и вычислительных устройств. - М .: «Наука», 1980. - 400с.

### **5. Основы программирования**

1. Стивенс Р. Алгоритмы. Теория и практическое применение / Р. Стивенс. – М.: Издательство «Э», 2016. – 544 с.
2. Дасгупта С. Алгоритмы / С. Дасгупта, Х. Пападимитриу, У. Вазирани. – М.: МЦНМО, 2014. – 320 с.
3. Шилдт Г. С# 4.0. Полное руководство / Г. Шилдт. – М.: Издательство «Вильямс», 2011. – 1056 с.
4. Нейгел К. С#5.0 и платформа .NET 4.5 для профессионалов / К. Нейгел, Б. Ивсен., Дж. Глини. – М.: Издательский дом «Вильямс», 2014. – 1440 с.
5. Ахо А. Структуры данных и алгоритмы / А. Ахо., Д. Хопкрофт, Д. Ульман. – М.: Издательский дом «Вильямс», 2001. – 384 с.

## **6. Программирование**

1. Страstrup Б. Язык программирования C++. 3-е изд./Пер. с англ.-СПб.; М.: «Невский Диалект» - «Издательство БИНОМ», 1999 г.-991 с., ил.
2. Шаммас Н.К. Основы C++ и объектно-ориентированного программирования. – К.: Диалектика, 1996. –448с., ил.

## **7. Сети и телекоммуникации**

1. Microsoft Corporation. Компьютерные сети. Учебный курс/Пер. с англ. – М.: Издательско-торговый дом «Русская Редакция», 2002. – 704стр.
2. Палмер М., Синклер Р.Б. Проектирование и внедрение компьютерных сетей. Учебный курс. – 2-е изд.. – СПб.: , 2004г. – 752с.
3. Спортак Марк. Компьютерные сети и сетевые технологии: Пер. с англ. – СПб.: ООО «ДиаСофтЮП», 2005. – 720 с.
4. Холме Дэн, Томас Орин. Управление и поддержка Microsoft Windows Server 2003. Учебный курс MCSA/MCSE/ Пер. с англ. – М.: Издательско-торговый дом «Русская редакция», 2004. – 448 стр.
5. Олифер В.Г., Олифер Н.А. Основы компьютерных сетей. – СПб.: Питер, 2009. – 352с.
6. Абрамов В.О., Клименко С.Ю. Базовые технологии компьютерных сетей: учебное пособи. - Киев. , 2011 -291с