МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М.Є. Жуковського  
«Харківський авіаційний інститут»

|  |  |  |
| --- | --- | --- |
|  |  | **ЗАТВЕРДЖЕНО** |
|  |  | вченою радою  Національного аерокосмічного університету ім. М.Є. Жуковського  «Харківський авіаційний інститут»  Заступник голови вченої ради \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  22 лютого 2017 р., протокол № 11 |

**ПРОГРАМА**

**ДОДАТКОВОГО ВСТУПНОГО ВИПРОБУВАННЯ**

для здобуття освітнього ступеня магістра

за освітньо-науковою програмою

зі спеціальності

**123 - Комп’ютерна інженерія**

(шифр та найменування)

(спеціалізація **Системне програмування**)

(найменування)

**у 2017 році**

Харків

2017

**ВСТУП**

Додаткове вступне випробування для здобуття освітнього ступеня магістра за освітньо-науковою програмою зі спеціальності

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_123 - Комп’ютерна інженерія \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(код та найменування)

(спеціалізації Системне програмування)

(найменування)

відбувається відповідно до «Правил прийому до Національного аерокосмічного університету ім. М.Є. Жуковського «Харківський авіаційний інститут» у 2017 році» у формі індивідуального письмового фахового іспиту, який приймає екзаменаційна комісія з певної спеціальності (спеціалізації), склад якої затверджується наказом ректора Університету.

До фахового іспиту входять питання за темами:

### Системне програмне забезпечення;

### Об’єктно-орієнтоване програмування;

### Архітектура комп’ютерів.

Перелік питань за темами наведений у програмі.

Результат додаткового критерію оцінювання знань фахового іспиту визначається за 100-бальною шкалою згідно з розділом VII, п. 2 Правил прийому до Національного аерокосмічного університету ім. М.Є. Жуковського «Харківський авіаційний інститут».

Іспит проводиться в вигляді тестів, що складається з 20 завдань з переліку питань, що входять до програми фахового випробування для здобуття освітнього ступеня магістра за освітньо-науковою програмою зі спеціальності 123-Комп’ютерна інженерія (спеціалізації Системне програмування ).

Кожне завдання тесту оцінюється в 5 балів від загальної кількості балів згідно правил прийому. Тест може містити в собі завдання в яких потрібно вибрати одну або декілька вірних відповідей (якщо це зазначається в умові завдання) з запропонованого переліку варіантів відповідей до кожного завдання. Для завдань в яких потрібно вибрати декілька варіантів відповідей бали будуть зараховані лише в тому випадку, якщо всі відповіді на запитання вибрані абітурієнтом будуть вірними. За виправлення відповіді в випадку якщо виправлена відповідь виявиться вірною абітурієнту знімається один або два бали в залежності від умов завдання (одна або декілька вірних відповідей).

1. **Питання за темою «Системне програмне забезпечення»**
2. Что такое конвейер в Linux? Для чего он нужен?
3. Что такое sha-bang? Для чего он используется?
4. Какие существуют команды для вывода содержимого файлов? В чём их различия?
5. Что такое GRUB? Из каких частей он состоит?
6. Что такое LILO? Из каких частей он состоит?
7. Дать описание системы sudo и её составляющих.
8. Дать описание поколениям ОС. К какому поколению относятся современные ОС?
9. Какой тип архитектуры у ядра Linux? Опишите его.
10. Дать описание программы dpkg.
11. Файловые системы в Linux.
12. Перечислите основные части Linux.
13. Каковы функции первичного загрузчика?
14. Каковы функции вторичного загрузчика?
15. Что такое UID? Какие UID соответствуют обычным пользователям?
16. Процесс init. Как он работает?
17. Команды Linux для работы с файлами и каталогами.
18. Что такое bash? Какие команды для проверки условий в нём используются?
19. Как вторичный загрузчик может прочитать файл с ядром? Описать несколько способов.
20. Что такое карта размещения?
21. Что такое процесс? Каким образом можно просмотреть состояние процессов и их активность "в реальном режиме времени"?
22. Типы процессов в Linux.
23. Команды управления процессами.
24. Какова особенность суперпользователя?
25. Состояния процесса.
26. Приоритеты для процессов. Какое значение nice соответствует наивысшему приоритету процесса?
27. Консольные текстовые редакторы в Linux. Описание и особенности.
28. Что такое дистрибутив Linux? Основные компоненты.
29. Что такое SWAP в Linux?
30. Типы файлов в Linux(помимо обычных файлов и каталогов).
31. Команды для работы с пользователями и группами пользователей.

Література

1. Мешков, В. Процессы в Linux [Текст] / В. Мешков – Москва, 2004 г. – С. 126.
2. Костромин, В. А. Основыработы в ОС Linux [Электронный ресурс] /

В. А. Костромин. – Интернет – Университет Информационных Технологий. – <http://www.intuit.ru/department/os/baseLinuxwork/>.

1. Соловьев, А. Разработка модулей ядра ОС Linux [Текст] / А. Соловьев —

Питер, 2012 г. – С. 85

Питання склав

К.т.н., доцент кафедри 503 Узун Д.Д.

(науковий ступень, посада) (ініціали та прізвище)

**2 Питання за темою "Об’єктно-орієнтоване програмування "**

1. Дайте определение понятию «исключение». Для чего используются исключения в программах на языке С#?
2. Дайте определение понятиям «класс» и «объект» с точки зрения ООП. Как взаимосвязаны эти понятия? Какие элементы могут быть членами классов?
3. Дайте определение понятию «инкапсуляция». Перечислите основные правила, которых необходимо придерживаться при определении доступа к членам класса.
4. Объясните понятие «конструктор». Для чего он используется? Разновидности конструкторов.
5. Дайте определение понятия «деконструктор». Когда и как вызывается деструктор?
6. Каким образом реализовано управление памятью в .NETFramework? Объясните понятия «куча» и «сборщик мусора». Каким образом можно оптимизировать работу сборщика мусора?
7. «Наследование» в ООП. Для чего оно используется.
8. Объясните связь понятий «полиморфизм» и «перегрузка» в отношении к классам, между которыми установлено отношение наследования. Назначение виртуальных методов.
9. Объясните значение и приведите пример объявления абстрактного класса.
10. Дайте определение понятию «виртуальный метод». Объясните, какие методы класса и почему не могут быть виртуальными.
11. Что такое «интерфейсы» в языке С#? Чем интерфейсы отличаются от классов и структур?
12. Сущность объектно-ориентированного программирования.
13. Особенности разработки программ для операционной системы Windows.

Событийно-ориентированный подход. Визуальное проектирование программ.

1. Общие сведения о платформе Microsoft.NET.
2. Языки программирования и компиляция на платформе .NET.
3. Интегрированная среда разработки (IDE) Visual Studio.NET.
4. Этапы разработки и режимы работы с Windows-приложением.
5. Проверка данных, вводимых пользователем.
6. Файлы и потоки.
7. Работа с файлами. Компонент OpenFileDialog.
8. Работа с файлами. Компонент SaveFileDialog.
9. Основные характеристики формы.
10. Приложение с несколькими формами.
11. Способы обмена данными между формами.
12. Работа MDI-приложения.
13. Модальное диалоговое окно и работа с ним.
14. Симметричные алгоритмы шифрования в .NET Framework.
15. Асимметричные алгоритмы шифрования в .NET Framework.
16. Проверка целостности данных при помощи хеширования.
17. Цифровая подпись данных.
18. Делегаты в .NET Framework.

Література

1. Шилдт, Герберт. С# 4.0: полное руководство [Текст] / Герберт Шилдт – Москва, 2011 г. – С. 1056.
2. Троелсен, Эндрю. Язык программирования C# 5.0 и платформа .NET 4.5 [Текст] / Эндрю Троелсен – Москва, 2013. – С. 1312.
3. Рихтер, Джеффри. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. [Текст] / Джеффри Рихтер – Питер, 2013. – 896 с.

Питання склав

к.т.н., доцент кафедри 503 Шостак А.В.

(науковий ступень, посада) (ініціали та прізвище)

**3 Питання за темою "Архітектура комп’ютерів "**

1. Архитектурные характеристики процессора x86.
2. Программная модель МП Intel x86. Назначение регистров. РОН. Адресные регистры.
3. Регистр флагов. Назначение флагов. Признаки результата. Флаги управления. Команды для работы с флагами.
4. Команды условного перехода по значению одного из признаков результата. Определение правильности результата.
5. Программирование на ассемблере. Формат оператора. Константы. Имена регистров. Синтаксис ассемблера Intel.
6. Система команд. Команды передачи данных.
7. Система команд. Простые арифметические команды.
8. Арифметика чисел различной длины. Команды передачи с увеличением разрядности.
9. Арифметика чисел различной длины. Команды преобразования чисел.
10. Команды расширенного умножения. Умножение чисел.
11. Команды расширенного деления чисел. Переполнение.
12. Команды управления программой. Команды безусловного перехода. Режимы адресации. Вычисление адреса.
13. Команда сравнения. Команды беззнакового условного перехода.
14. Команда сравнения. Команды знакового условного перехода.
15. Логические команды. Команда НЕ. Логическая операция И.
16. Логическая операция ИЛИ. Логическая операция Исключающее ИЛИ.
17. Операции с битами. Установка битов. Сброс битов.
18. Операции с битами. Инвертирование битов. Сброс битов.
19. Сдвиги. Параметры сдвигов. Формат команд сдвигов. Логические сдвиги.
20. Сдвиги. Параметры сдвигов. Формат команд сдвигов. Арифметический сдвиг.
21. Сдвиги. Параметры сдвигов. Формат команд сдвигов. Циклические сдвиги. Применение.
22. Сдвиг числа двойной длины на несколько разрядов. Команды сдвига числа на несколько разрядов.
23. Умножение беззнаковых чисел на константу. Методика. Пример.
24. Умножение знаковых чисел на константу. Методика. Пример.
25. Упаковка и распаковка битовых групп. Операция упаковки. Методика. Пример.
26. Упаковка и распаковка битовых групп. Операция распаковки. Методика. Пример.
27. Операции с числами многократной точности. Адресация чисел многократной точности. Сложение и вычитание чисел многократной точности. Методика. Примеры.
28. Операции с числами многократной точности. Адресация чисел многократной точности. Сравнение. Изменение знака чисел многократной точности. Методика. Примеры.
29. Операции с числами многократной точности. Адресация чисел многократной точности. Сдвиги чисел многократной точности. Методика. Примеры.
30. Формат команды. Компоненты машинного кода инструкции. Префикс команды. Виды префиксов. Префикс замены сегмента.

Література

1. Пирогов, В.Ю. Assembler. Учебный курс [Текст] / В.Ю. Пирогов – Москва, 2001г. – С. 848.
2. Кучеренко, В. Ассемблер. Тонкости, хитрости и секреты программирования [Текст] / В. Кучеренко – Москва, 2001. – С. 213.
3. Крупник, А. Изучаем Ассемблер. [Текст] / А. Крупник – Питер, 2013. – 256 с.

Питання склав

к.т.н., доцент кафедри 503 Дужий В.И.

(науковий ступень, посада) (ініціали та прізвище)

Завідувач кафедри 503 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.С.Харченко

(підпис) (ініціали та прізвище)

Програму розглянуто й узгоджено на випусковій кафедрі 503\_\_\_\_\_\_\_\_

Протокол № 8 від «20» січня 2017 р.

Програму додаткового вступного випробування для здобуття освітнього ступеня магістра за освітньо-науковою програмою зі спеціальності 123 - Комп’ютерна інженерія (спеціалізації Системне програмування) узгоджено науково-методичною комісією Національного аерокосмічного університету ім. М.Є. Жуковського «Харківський авіаційний інститут» з галузей знань «Математика та статистика», «Інформаційні технології», «Автоматизація та приладобудування», «Хімічна біоінженерія», «Електроніка та телекомунікації» (НМК 2).

Протокол № \_\_\_ від \_\_\_\_\_\_\_ лютого 2017 р.

Голова НМК 2

к.т.н., доц. О.В. Заболотний

погоджено науково-методичною комісією Національного аерокосмічного університету ім. М.Є. Жуковського «Харківський авіаційний інститут»

Протокол № \_\_\_ від \_\_\_\_\_\_\_лютого 2017 р.

Голова НМК Університету

д.т.н., проф. В.М. Павленко