

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М.С. Жуковського
«Харківський авіаційний інститут»

ЗАТВЕРДЖЕНО

вченою радою
Національного аерокосмічного
університету ім. М.С. Жуковського
«Харківський авіаційний інститут»

Заступник Голови вченої ради
О.В. Гайдачук

«21» лютого 2018 р. протокол № 7



**ПРОГРАМА
ДОДАТКОВОГО ВСТУПНОГО ВИПРОБУВАННЯ**

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

123 - Комп'ютерна інженерія
(код та найменування)

(освітня програма **Комп'ютерні системи та мережі,
Системне програмування,
Програмовні мобільні системи та Інтернет речей**)
(найменування)

у 2018 році

Харків
2018

ВСТУП

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

123 - Комп'ютерна інженерія

(код та найменування)

(освітня програма Комп'ютерні системи та мережі, Системне програмування, Програмовні мобільні системи та Інтернет речей)

(найменування)

відбувається відповідно до «Правил прийому на навчання до Національного аерокосмічного університету імені М.Є. Жуковського «Харківський авіаційний інститут» в 2018 році» у формі індивідуального письмового фахового іспиту, який приймає екзаменаційна комісія з певної спеціальності (освітньої програми), 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.

Критерії оцінювання знань

1. Результат додаткового фахового іспиту визначається за 100-бальною шкалою. При отриманні вступником 60 балів та більше він допускається до вступного випробування.

2. 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 (освітня програма Комп'ютерні системи та мережі, Системне програмування, Програмовні мобільні системи та Інтернет речей).

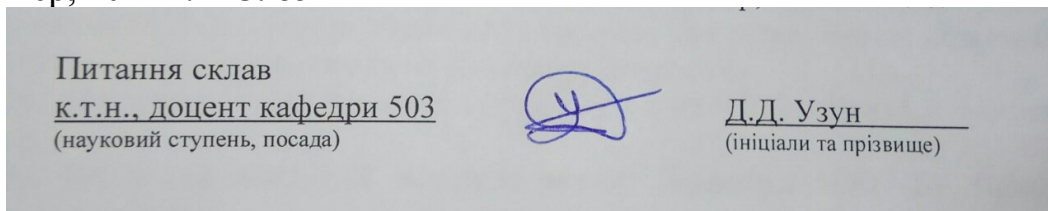
Кожне завдання тесту оцінюється в 5 балів від загальної кількості балів згідно правил прийому. Тест може містити в собі завдання в яких потрібно вибрати одну або декілька вірних відповідей (якщо це зазначається в умові завдання) з запропонованого переліку варіантів відповідей до кожного завдання. Для 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»

- 1.1. Що таке конвейер в Linux? Для чого он нужен?
- 1.2. Що таке sha-bang? Для чого он використовується?
- 1.3. Які існують команди для виводу вмісту файлів? В чому їх різниця?
- 1.4. Що таке GRUB? З яких частин він складається?
- 1.5. Що таке LILO? З яких частин він складається?
- 1.6. Дати опис системи sudo і її складових.
- 1.7. Дати опис поколінь ОС. До якого покоління належать сучасні ОС?
- 1.8. Якого типу архітектури у ядра Linux? Опишіть його.
- 1.9. Дати опис програми dpkg.
- 1.10. Файлові системи в Linux.
- 1.11. Перерахуйте основні частини Linux.
- 1.12. Які функції первинного завантажувача?
- 1.13. Які функції вторинного завантажувача?
- 1.14. Що таке UID? Які UID відповідають звичайним користувачам?
- 1.15. Процес init. Як він працює?
- 1.16. Команди Linux для роботи з файлами і каталогами.
- 1.17. Що таке bash? Які команди для перевірки умов в ньому використовуються?
- 1.18. Як вторинний завантажувач може прочитати файл з ядром? Описати кілька способів.
- 1.19. Що таке карта розміщення?
- 1.20. Що таке процес? Яким чином можна переглянути стан процесів і їх активність "в реальному режимі часу"?
- 1.21. Типи процесів в Linux.
- 1.22. Команди управління процесами.
- 1.23. Яка особливість суперкористувача?
- 1.24. Стани процесу.
- 1.25. Пріоритети для процесів. Яке значення nice відповідає найвищому пріоритету процесу?
- 1.26. Консольні текстові редактори в Linux. Опис і особливості.
- 1.27. Що таке дистрибутив Linux? Основні компоненти.
- 1.28. Що таке SWAP в Linux?
- 1.29. Типи файлів в Linux(окрім звичайних файлів і каталогів).
- 1.30. Команди для роботи з користувачами і групами користувачів.

Література

1. Мешков, В. Процессы в Linux [Текст] / В. Мешков – Москва, 2004 г. – С. 126.
2. Костромин, В. А. Основы работы в ОС Linux [Электронный ресурс] / В. А. Костромин. – Интернет – Университет Информационных Технологий. – <http://www.intuit.ru/department/os/baseLinuxwork/>.
3. Соловьев, А. Разработка модулей ядра ОС Linux [Текст] / А. Соловьев — Питер, 2012 г. – С. 85



2 Питання за темою "Об'єктно-орієнтоване програмування "

- 2.1. Дайте определение понятию «исключение». Для чего используются исключения в программах на языке C#?
- 2.2. Дайте определение понятиям «класс» и «объект» с точки зрения ООП. Как взаимосвязаны эти понятия? Какие элементы могут быть членами классов?
- 2.3. Дайте определение понятию «инкапсуляция». Перечислите основные правила, которых необходимо придерживаться при определении доступа к членам класса.
- 2.4. Объясните понятие «конструктор». Для чего он используется? Разновидности конструкторов.
- 2.5. Дайте определение понятия «деконструктор». Когда и как вызывается деструктор?
- 2.6. Каким образом реализовано управление памятью в .NETFramework? Объясните понятия «куча» и «сборщик мусора». Каким образом можно оптимизировать работу сборщика мусора?
- 2.7. «Наследование» в ООП. Для чего оно используется.
- 2.8. Объясните связь понятий «полиморфизм» и «перегрузка» в отношении к классам, между которыми установлено отношение наследования. Назначение виртуальных методов.
- 2.9. Объясните значение и приведите пример объявления абстрактного класса.
- 2.10. Дайте определение понятию «виртуальный метод». Объясните, какие методы класса и почему не могут быть виртуальными.
- 2.11. Что такое «интерфейсы» в языке C#? Чем интерфейсы отличаются от классов и структур?
- 2.12. Сущность объектно-ориентированного программирования.
- 2.13. Особенности разработки программ для операционной системы Windows. Событийно-ориентированный подход. Визуальное проектирование программ.

- 2.14. Общие сведения о платформе Microsoft.NET.
- 2.15. Языки программирования и компиляция на платформе .NET.
- 2.16. Интегрированная среда разработки (IDE) Visual Studio.NET.
- 2.17. Этапы разработки и режимы работы с Windows-приложением.
- 2.18. Проверка данных, вводимых пользователем.
- 2.19. Файлы и потоки.
- 2.20. Работа с файлами. Компонент OpenFileDialog.
- 2.21. Работа с файлами. Компонент SaveFileDialog.
- 2.22. Основные характеристики формы.
- 2.23. Приложение с несколькими формами.
- 2.24. Способы обмена данными между формами.
- 2.25. Работа MDI-приложения.
- 2.26. Модальное диалоговое окно и работа с ним.
- 2.27. Симметричные алгоритмы шифрования в .NET Framework.
- 2.28. Асимметричные алгоритмы шифрования в .NET Framework.
- 2.29. Проверка целостности данных при помощи хеширования.
- 2.30. Цифровая подпись данных.
- 2.31. 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 С# 5.0 и платформа .NET 4.5 [Текст] / Эндрю Троелсен – Москва, 2013. – С. 1312.
3. Рихтер, Джеффри. CLR via С#. Программирование на платформе Microsoft .NET Framework 4.5 на языке С#. [Текст] / Джеффри Рихтер – Питер, 2013. – 896 с.

Питання склав
к.т.н., доцент кафедри 503
(науковий ступень, посада)



Шостак А.В.
(ініціали та прізвище)

3 Питання за темою "Архітектура комп'ютерів "

- 3.1. Архитектурные характеристики процессора x86.
- 3.2. Программная модель МП Intel x86. Назначение регистров. РОН. Адресные регистры.

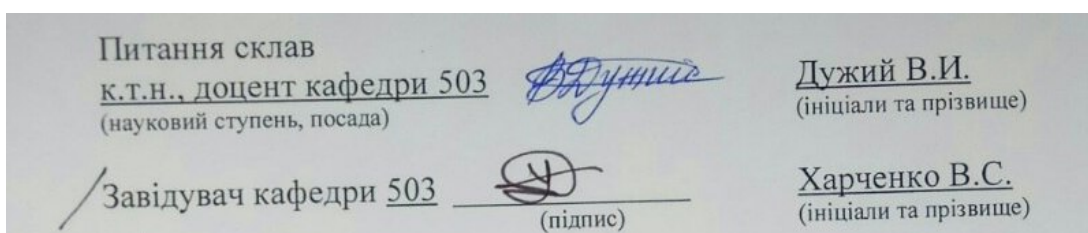
- 3.3. Регистр флагов. Назначение флагов. Признаки результата. Флаги управления. Команды для работы с флагами.
- 3.4. Команды условного перехода по значению одного из признаков результата. Определение правильности результата.
- 3.5. Программирование на ассемблере. Формат оператора. Константы. Имена регистров. Синтаксис ассемблера Intel.
- 3.6. Система команд. Команды передачи данных.
- 3.7. Система команд. Простые арифметические команды.
- 3.8. Арифметика чисел различной длины. Команды передачи с увеличением разрядности.
- 3.9. Арифметика чисел различной длины. Команды преобразования чисел.
- 3.10. Команды расширенного умножения. Умножение чисел.
- 3.11. Команды расширенного деления чисел. Переполнение.
- 3.12. Команды управления программой. Команды безусловного перехода. Режимы адресации. Вычисление адреса.
- 3.13. Команда сравнения. Команды беззнакового условного перехода.
- 3.14. Команда сравнения. Команды знакового условного перехода.
- 3.15. Логические команды. Команда НЕ. Логическая операция И.
- 3.16. Логическая операция ИЛИ. Логическая операция Исключающее ИЛИ.
- 3.17. Операции с битами. Установка битов. Сброс битов.
- 3.18. Операции с битами. Инвертирование битов. Сброс битов.
- 3.19. Сдвиги. Параметры сдвигов. Формат команд сдвигов. Логические сдвиги.
- 3.20. Сдвиги. Параметры сдвигов. Формат команд сдвигов. Арифметический сдвиг.
- 3.21. Сдвиги. Параметры сдвигов. Формат команд сдвигов. Циклические сдвиги. Применение.
- 3.22. Сдвиг числа двойной длины на несколько разрядов. Команды сдвига числа на несколько разрядов.
- 3.23. Умножение беззнаковых чисел на константу. Методика. Пример.
- 3.24. Умножение знаковых чисел на константу. Методика. Пример.
- 3.25. Упаковка и распаковка битовых групп. Операция упаковки. Методика. Пример.
- 3.26. Упаковка и распаковка битовых групп. Операция распаковки. Методика. Пример.
- 3.27. Операции с числами многократной точности. Адресация чисел многократной точности. Сложение и вычитание чисел многократной точности. Методика. Примеры.
- 3.28. Операции с числами многократной точности. Адресация чисел многократной точности. Сравнение. Изменение знака чисел многократной точности. Методика. Примеры.

3.29. Операції с числами многократной точности. Адресация чисел многократной точности. Сдвиги чисел многократной точности. Методика.

3.30. Формат команды. Компоненты машинного кода инструкции. Префикс команды. Види 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.



Програму розглянуто й узгоджено на випусковій кафедрі 503
Протокол № 8 від «06» 02 2018 р.

Програму вступного випробування для здобуття освітнього ступеня магістра за освітньо-науковою програмою зі спеціальності 123 - Комп'ютерна інженерія (освітня програма Комп'ютерні системи та мережі, Системне програмування, Програмовні мобільні системи та Інтернет речей).

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 та біоінженерія», «Електроніка та телекомунікації» (НМК 2)»

Протокол № 1 від 08 лютого 2018 р.

Голова НМК 2
к.т.н., доц.

О.В. Заболотний