

РЕПУБЛИКА БЪЛГАРИЯ

Министър на образованието и науката

## З А П О В Е Д

**№ РД 09-4604/24.11.2021 г.**

На основание чл. 36, ал. 2 от Закона за професионалното образование и обучение, във връзка с чл. 2, ал. 1 и 2 от Наредба № 1 от 19.02.2020 г. за организацията и провеждането на изпитите за придобиване на професионална квалификация, при спазване изискванията на чл. 66, ал. 1 и 2 от Административнопроцесуалния кодекс

## У Т В Ъ Р Ж Д А В А М

Национална изпитна програма за провеждане на държавен изпит за придобиване на трета степен на професионална квалификация за специалност **код 4810301 „Приложно програмиране“** от професия **код 481030 „Приложен програмист“** от професионално направление **код 481 „Ком- пютърни науки“** съгласно приложението.

X

АКАД. НИКОЛАЙ ДЕНКОВ

Министър на образованието и науката

Приложение

## МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО И НАУКАТА

**НАЦИОНАЛНА ИЗПИТНА ПРОГРАМА**

# ЗА ПРОВЕЖДАНЕ

**НА ДЪРЖАВЕН ИЗПИТ ЗА ПРИДОБИВАНЕ**

# НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ

|  |  |  |
| --- | --- | --- |
|  | **Код**  **по СППОО** | **Наименование** |
| **ПРОФЕСИОНАЛНО НАПРАВЛЕНИЕ** | **481** | **Компютърни науки** |
| **ПРОФЕСИЯ** | **481030** | **Приложен програмист** |
| **СПЕЦИАЛНОСТ** | **4810301** | **Приложно програмиране** |

## Утвърдена със Заповед № РД 09-4604/24.11.2021 г.

София, 2021 г.

## І. ПРЕДНАЗНАЧЕНИЕ И ЦЕЛ НА ИЗПИТНАТА ПРОГРАМА

Националната изпитна програма е предназначена за провеждане на държавния изпит за при- добиване на трета степен на професионална квалификация по специалност **код 4810301 „При- ложно програмиране“**, професия **код 481030 „Приложен програмист“** от Списъка на профе- сиите за професионално образование и обучение по чл. 6 от Закона за професионалното образо- вание и обучение (ЗПОО).

Целта на настоящата изпитна програма е да определи единни критерии за оценка на профе- сионалните компетентности на обучаваните, изискващи се за придобиване на трета степен по изучаваната **професия „Приложен програмист“**, **специалност „Приложно програмиране“**.

Националната изпитна програма е разработена във връзка с чл. 36 от ЗПОО и чл. 2, ал. 1 и ал. 2 от Наредба № 1 от 19.02.2020 г. за организацията и провеждането на изпитите за придоби- ване на професионална квалификация.

## ІІ. СЪДЪРЖАНИЕ НА НАЦИОНАЛНАТА ИЗПИТНА ПРОГРАМА

За държавен изпит за придобиване на професионална квалификация:

## Част по теория на професията

* 1. Изпитни теми с кратко описание на учебното съдържание по всяка тема
  2. Критерии за оценяване на резултатите от обучението по всяка изпитна тема
  3. Матрица на писмен тест по всяка изпитна тема
  4. Критерии и показатели за оценяване на дипломния проект и неговата защита

## Част по практика на професията

* 1. указание за съдържанието на индивидуалните задания
  2. критериите за оценяване на резултатите от обучението

## Система за оценяване

1. **Препоръчителна литература**

## Приложения

а. Изпитен билет – част по теория на професията б. Индивидуално задание по практика

в. Указание за разработване на писмен тест

г. Индивидуално задание за разработване на дипломен проект

д. Указания за съдържанието и оформянето на дипломния проект е. рамка на рецензия на дипломен проект.

## ІIІ. ДЪРЖАВЕН ИЗПИТ ЗА ПРИДОБИВАНЕ НА ПРОФЕСИОНАЛНА КВАЛИФИ- КАЦИЯ – ЧАСТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА

1. **Изпитни теми, критерии за оценяване на резултатите и матрица на писмен тест по всяка изпитна тема**

*Изпитна тема № 1:* **Изчисления, линейни, разклонени и циклични алгоритми**

Дефиниция на основни понятия: програмиране, език за програмиране, среда за разработка (IDE), компилация и интерпретация. Променливи и типове от данни – дефиниция, сравнение на типове и употреба. Пресмятания, аритметични и логически оператори, изрази – видове и разработка. Описание на условни конструкции – пълна и кратка форма. Логически изрази и оператори за сравнение – видове и разработка. Оператор за многовариантен избор – описание, сравнение с условен оператор. Циклични оператори – описание и разработка. Видове цикли – сравнение, при- лики и разлики. Същност на подпрограми (функции/методи), видове параметри, видове връщана стойност, предимства при употребата на подпрограми. Определяне на реда на изпълнение на фрагментите в кода (program flow). Работа със среда за разработка (IDE) – създаване и зареждане на проект в среда за разработка. Създаване и/или поправка/допълване на вече съществуващи ком- пютърни програми, решаващи изчислителни и алгоритмични задачи със средствата на програмен език.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 1** | **Максимален брой точки** |
| 1. Дефинира понятията: програмиране, език за програмиране, среда за разра- ботка (IDE), компилация и интерпретация. Демонстрира употребата на ос- новни функционалности на среда за разработка (създаване и зареждане на проект, стартиране на проект). | 12 |
| 2. Описва понятието променлива. Сравнява типове променливи. Разработва програми/програмни фрагменти с аритметични и логически изрази с участи- ето на променливи и числа. | 12 |
| 3. Описва условни конструкции – пълна и кратка форма. Описва оператор за многовариантен избор (switch). Сравнява условни конструкции с няколко ус- ловия (else if) и оператор за многовариантен избор (switch). Определя кои фрагменти от код се изпълняват, колко пъти и в какъв ред при условен опера- тор и/или многовариантен избор. | 16 |
| 4. Описва и разработва програми/програмни фрагменти с оператори за цикли. Различава операторите за цикли. Определя и посочва кои фрагменти код се изпълняват, колко пъти и в какъв ред при оператори за цикли. | 20 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 1** | **Максимален брой точки** |
| 5. Обяснява същността на подпрограмите (функции/методи). Посочва видове па- раметри и връщани стойности. Дава пример за предимствата от използването на методи. | 16 |
| 6. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 24 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 1 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира понятията: програмиране, език за програмиране, среда за разработка (IDE), компилация и интерпретация. Де- монстрира употребата на основни функционалности на среда за разработка (създаване и зареждане на проект, стартиране на проект). | **12** | 3 |  | 1 |  |
| 2. Описва понятието променлива. Сравнява типове промен- ливи. Разработва програми/програмни фрагменти с аритме- тични и логически изрази с участието на променливи и числа. | **12** | 1 | 1 | 1 |  |
| 3. Описва условни конструкции – пълна и кратка форма. Описва оператор за многовариантен избор (switch). Сравнява ус- ловни конструкции с няколко условия (else if) и оператор за многовариантен избор (switch). Определя кои фрагменти от код се изпълняват, колко пъти и в какъв ред при условен опе- ратор и/или многовариантен избор. | **16** | 2 | 1 |  | 1 |
| 4. Описва и разработва програми/програмни фрагменти с опе- ратори за цикли. Различава операторите за цикли. Определя и посочва кои фрагменти код се изпълняват, колко пъти и в какъв ред при оператори за цикли. | **20** | 1 | 1 | 1 | 1 |
| 5. Обяснява същността на подпрограмите (функции/методи). Посочва видове параметри и връщани стойности. Дава при- мер за предимствата от използването на методи. | **16** | 2 | 1 |  | 1 |
| 6. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **24** |  |  |  | 3 |
| **Общ брой задачи:** | **22** | **9** | **4** | **3** | **6** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 1 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **Общ брой точки:** | **100** | **18** | **16** | **18** | **48** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 2:* **Обработка на колекции от данни**

Сорс-контрол системи – дефиниция, видове (централизирани и децентрализирани) и команди за работа. Видове типове данни. Понятие за обект. Видове бройни системи, операции и преобразу- ване от една бройна система в друга. Изчисления в бройни системи. Едномерни и многомерни масиви – дефиниция, деклариране, описание и илюстрация на структурата, разработка на алго- ритми върху масиви. Списъци – дефиниция, деклариране, описание на структурата, основни опе- рации, разработка на алгоритми върху списъци, изводи за предимствата и недостатъците спрямо масивите. Дебъгване и работа с дебъгер – откриване на проблеми в програми с използването на дебъгер. Символни низове – дефиниция, деклариране, методи за работа, разработка на алгоритми за обработка на текст (извличане на подниз, замяна на низ и др.). Речници (хеш-таблици) – дефи- ниция, деклариране, устройство, методи за работа, решаване на задачи върху речници, разлика между ключ и стойност. Определяне на реда на изпълнение на фрагментите в кода (program flow). Създаване и/или поправка/допълване на вече съществуващи компютърни програми, решаващи изчислителни и алгоритмични задачи със средствата на програмен език.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 2** | **Максимален брой точки** |
| 1. Дефинира понятието сорс-контрол система. Изброява команди за работа със сорс-контрол системите. Различава централизирани и децентрализирани сорс- контрол системи. | 8 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 2** | **Максимален брой точки** |
| 2. Демонстрира откриване и отстраняване на проблем в програма с помощта на дебъгера. | 6 |
| 3. Описва типове данни. Дефинира понятието обект. Посочва видове бройни системи. Преобразува числа от една бройна система в друга и изчислява из- рази с тях. | 16 |
| 4. Дефинира понятието едномерен и многомерен масив. Описва декларирането на масиви. Илюстрира графично едномерни и многомерни масиви. Разработва алгоритми върху масиви. | 16 |
| 5. Дефинира понятието списък. Описва декларирането на списъци. Дава пример за основните операции със списъци. Прави заключения за предимствата и не- достатъците при употребата на списъци спрямо масиви. | 14 |
| 6. Дефинира понятието символен низ. Описва декларирането на символен низ. Посочва методи за работа със символни низове от изучаван език за програми- ране. Разработва програми за алгоритми за обработка на текст чрез операции за текстови низове (извличане на подниз, замяна на низ и др.). | 10 |
| 7. Дефинира понятието речник. Описва устройството на речник (хеш-таблица). Посочва методи за работа с речник. Решава задачи с използването на подхо- дящи методи върху речник. Различава ключ и стойност в речника. | 14 |
| 8. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 16 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 2 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира понятието сорс-контрол система. Изброява ко- манди за работа със сорс-контрол системите. Различава цен- трализирани и децентрализирани сорс-контрол системи. | **8** | 2 | 1 |  |  |
| 2. Демонстрира откриване и отстраняване на проблем в прог- рама с помощта на дебъгера. | **6** |  |  | 1 |  |
| 3. Описва типове данни. Дефинира понятието обект. Посочва видове бройни системи. Преобразува числа от една бройна система в друга и изчислява изрази с тях. | **16** | 3 | 1 | 1 |  |
| 4. Дефинира понятието едномерен и многомерен масив. Описва декларирането на масиви. Илюстрира графично едномерни и многомерни масиви. Разработва алгоритми върху масиви. | **16** | 1 |  | 1 | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 2 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 5. Описва декларирането на списъци. Дава пример за основните операции със списъци. Прави заключения за предимствата и недостатъците при употребата на списъци спрямо масиви. | **14** | 1 | 1 |  | 1 |
| 6. Описва декларирането на символен низ. Посочва методи за работа със символни низове от изучаван език за програми- ране. Разработва програми за алгоритми за обработка на текст чрез операции за текстови низове (извличане на подниз, замяна на низ и др.). | **10** | 2 |  | 1 |  |
| 7. Дефинира понятието речник. Описва устройството на речник (хеш-таблица). Посочва методи за работа с речник. Решава задачи с използването на подходящи методи върху речник. Различава ключ и стойност в речника. | **14** | 3 | 2 |  |  |
| 8. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **16** |  |  |  | 2 |
| **Общ брой задачи:** | **25** | **12** | **5** | **4** | **4** |
| **Общ брой точки:** | **100** | **24** | **20** | **24** | **32** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 3:* **Представяне на обекти от реалния свят с програмен код**

Понятия: Клас, конструктор, поле и свойство. Създаване на обекти от клас. Функции/методи в класовете, тип и стойност на връщане, параметри и видове параметри на функция/метод – упот- реба. Необходимост от употреба на ключовата дума this. Енкапсулация на данни в класовете, методи за достъп и промяна на полета (getters/setters), модификатори за достъп. Статични полета и методи в класовете. Създаване и/или поправка/допълване на вече съществуващи компютърни програми, описващи обекти от реалния свят със средствата на програмен език.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 3** | **Максимален брой точки** |
| 1. Дефинира: клас, конструктор, поле, свойство. Описва създаване на обекти от клас. Различава понятията конструктор, метод и свойства. | 8 |
| 2. Дефинира понятията функции/методи, тип и стойност на връщане, параметри и посочва видове параметри на функция/метод. Демонстрира дефинирането и употребата на функции/методи. | 8 |
| 3. Прави заключения и изводи за употребата на ключовата дума this и дава при- мер за ситуации, които показват необходимостта от употребата на ключовата дума this. | 12 |
| 4. Обяснява енкапсулирането на данни в класовете. Дава пример за употребата на методите за достъп и промяна на енкапсулираните данни. Открива и моди- фицира (ако е необходимо) модификаторите за достъп на полета, свойства, методи, за да бъде спазена правилната енкапсулация на данните. | 14 |
| 5. Дефинира и описва статичен клас и статични членове на класа. Дава пример за употребата на статични членове в клас. | 6 |
| 6. Дефинира по-сложни класове. Различава модификатори за достъп. Разработва по-сложни класове с правилна енкапсулация на членовете на класа. | 12 |
| 7. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 40 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 3 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира: клас, конструктор, поле, свойство. Описва създа- ване на обекти от клас. Различава понятията конструктор, ме- тод и свойства. | **8** | 2 | 1 |  |  |
| 2. Дефинира понятията функции/методи, тип и стойност на връ- щане, параметри и посочва видове параметри на функция/ме- тод. Демонстрира дефинирането и употребата на функ- ции/методи. | **8** | 1 |  | 1 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 3 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 3. Прави заключения и изводи за употребата на ключовата дума this и дава пример за ситуации, които показват необходи- мостта от употребата на ключовата дума this. | **12** |  | 1 |  | 1 |
| 4. Обяснява енкапсулирането на данни в класовете. Дава при- мер за употребата на методите за достъп и промяна на енкап- сулираните данни. Открива и модифицира (ако е необхо- димо) модификаторите за достъп на полета, свойства, ме- тоди, за да бъде спазена правилната енкапсулация на данните. | **14** |  | 2 | 1 |  |
| 5. Дефинира и описва статичен клас и статични членове на класа. Дава пример за употребата на статични членове в клас. | **6** | 1 | 1 |  |  |
| 6. Дефинира по-сложни класове. Различава модификатори за достъп. Разработва по-сложни класове с правилна енкапсула- ция на членовете на класа. | **12** | 1 | 1 | 1 |  |
| 7. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **40** |  |  |  | 5 |
| **Общ брой задачи:** | **20** | **5** | **6** | **3** | **6** |
| **Общ брой точки:** | **100** | **10** | **24** | **18** | **48** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 4:* **Обектно-ориентирано програмиране**

Концепцията за типизиране на класове. Създаване на шаблонни класове и методи. Наследяване на класове. Презаписване и презареждане. Абстракция и интерфейси – връзка и разлики, упот- реба на абстрактни класове и интерфейси. Полиморфизъм – понятие, видове полиморфизъм. Принципи на обектно-ориентираното програмиране – примери и приложение. Създаване и/или поправка/допълване на вече съществуващи компютърни програми, описващи обекти от реалния свят със средствата на програмен език.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 4** | **Максимален брой точки** |
| 1. Познава концепцията за типизиране на класове чрез шаблонни класове и ме- тоди. Демонстрира създаването и употребата на шаблонни класове и методи. | 8 |
| 2. Описва и обяснява концепцията за наследяване на класове. Демонстрира нас- ледяването на класове. | 12 |
| 3. Различава презаписване (override) и презареждане (overload) на метод. | 4 |
| 4. Посочва принципите на обектно-ориентираното програмиране и дава примери за приложението им. | 12 |
| 5. Описва абстрактни класове и интерфейси. Различава абстрактен клас и интер- фейс. Демонстрира и прави заключения и изводи за употребата на интерфейси и абстрактни класове. | 26 |
| 6. Дефинира понятието полиморфизъм и различава видовете полиморфизъм. | 6 |
| 7. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 32 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 4 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Познава концепцията за типизиране на класове чрез шаб- лонни класове и методи. Демонстрира създаването и употре- бата на шаблонни класове и методи. | **8** | 1 |  | 1 |  |
| 2. Описва и обяснява концепцията за наследяване на класове. Демонстрира наследяването на класове. | **12** | 1 | 1 | 1 |  |
| 3. Различава презаписване (override) и презареждане (overload) на метод. | **4** |  | 1 |  |  |
| 4. Посочва принципите на обектно-ориентираното програми- ране и дава примери за приложението им. | **12** | 2 | 2 |  |  |
| 5. Описва абстрактни класове и интерфейси. Различава абстрак- тен клас и интерфейс. Демонстрира и прави заключения и из- води за употребата на интерфейси и абстрактни класове. | **26** | 1 | 1 | 2 | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 4 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 6. Дефинира понятието полиморфизъм и различава видовете полиморфизъм. | **6** | 1 | 1 |  |  |
| 7. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **32** |  |  |  | 4 |
| **Общ брой задачи:** | **21** | **6** | **6** | **4** | **5** |
| **Общ брой точки:** | **100** | **12** | **24** | **24** | **40** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 5:* **Реализиране на собствени линейни структури от данни**

Основните алгоритми, използвани в практиката: рекурсия и рекурсивни алгоритми, алгоритми за търсене (линейно, двоично, интерполационно) и сортиране (метод на мехурчето, метод на пря- ката селекция, сортиране чрез вмъкване и др.) – видове, разлики между алгоритми за търсене и сортиране, имплементация на алгоритми за търсене и сортиране, разработка на рекурсивни алго- ритми. Линейна структура от данни – понятие. Статична и динамична реализация на имплемен- тация на структури от данни – разтеглив масив, свързан списък, двойно свързан списък, стек и опашка. Създаване и/или поправка/допълване на вече съществуващи компютърни програми, описващи линейни структури от данни, рекурсивни, търсещи или сортиращи алгоритми със сред- ствата на програмен език.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 5** | **Максимален брой точки** |
| 1. Дефинира понятието линейна структура от данни. Различава алгоритми за търсене и сортиране. | 6 |
| 2. Посочва видове алгоритми за търсене. Демонстрира имплементация на избран алгоритъм за търсене. | 8 |
| 3. Различава статична и динамична реализация на линейна структура от данни. Демонстрира имплементация на статична и динамична реализация на избрана линейна структури от данни – разтеглив масив, свързан списък, двойно свър- зан списък, реализация на стек чрез разтеглив масив и свързан списък, реали- зация на опашка чрез разтеглив масив и свързан списък. Прави заключения за бързодействието на операциите при статична и динамична реализация. | 18 |
| 4. Описва и дава примери за основните алгоритми за сортиране – сортиране чрез пряка селекция, сортиране чрез метода на мехурчето, сортиране чрез вмъкване и други сортиращи алгоритми. Демонстрира имплементацията на избрани ал- горитми за сортиране. | 18 |
| 5. Обяснява понятието рекурсия и разработва програми с използването на рекур- сивни алгоритми. | 10 |
| 6. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 40 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 5 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира понятието линейна структура от данни. Различава алгоритми за търсене и сортиране. | **6** | 1 | 1 |  |  |
| 2. Посочва видове алгоритми за търсене. Демонстрира импле- ментация на избран алгоритъм за търсене. | **8** | 1 |  | 1 |  |
| 3. Различава статична и динамична реализация на линейна структура от данни. Демонстрира имплементация на ста- тична и динамична реализация на избрана линейна структури от данни – разтеглив масив, свързан списък, двойно свързан списък, реализация на стек чрез разтеглив масив и свързан списък, реализация на опашка чрез разтеглив масив и свързан списък. Прави заключения за бързодействието на операциите при статична и динамична реализация. | **18** |  | 1 | 1 | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 5 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 4. Описва и дава примери за основните алгоритми за сортиране  – сортиране чрез пряка селекция, сортиране чрез метода на мехурчето, сортиране чрез вмъкване и други сортиращи ал- горитми. Демонстрира имплементацията на избрани алго- ритми за сортиране. | **18** | 1 | 1 | 2 |  |
| 5. Обяснява понятието рекурсия и разработва програми с изпол- зването на рекурсивни алгоритми. | **10** |  | 1 | 1 |  |
| 6. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **40** |  |  |  | 5 |
| **Общ брой задачи:** | **18** | **3** | **4** | **5** | **6** |
| **Общ брой точки:** | **100** | **6** | **16** | **30** | **48** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 6:* **Бази данни**

Въведение в базите от данни. Типове данни. Основни команди. Основни SQL команди за работа с бази данни – създаване, промяна и премахване на бази от данни, таблици и колони. Моделиране на релационни бази от данни – нормализация и нормални форми. Типове връзки, каскадни опе- рации и ограничения. Заявки за извличане и промяна на данни (SELECT, INSERT, UPDATE, DELETE). Сложни заявки за извличане на данни. Съединения на таблици (SQL JOIN). Агреги- ращи функции и групиране на данни. Филтриране и сортиране на групирани данни. Функции, транзакции и съхранени процедури.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и под- ходяща версия на СУБД от изучаваните. Например MSSQL, MySQL, PostgreSQL, Oracle Database и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 6** | **Максимален брой точки** |
| 1. Дефинира понятието бази данни и различава типовете данни. | 6 |
| 2. Демонстрира знанията си за създаване, промяна и премахване на таблици в бази данни. | 6 |
| 3. Дефинира процеса на нормализация и диференцира нормалните форми. | 10 |
| 4. Посочва, различава и демонстрира типовете връзки в релационните бази данни. | 14 |
| 5. Прави изводи за различните каскадни операции. | 8 |
| 6. Познава, обяснява и дава пример за ограничители (constraints). | 16 |
| 7. Познава, разбира и различава SQL заявки за основните операции (CRUD). | 16 |
| 8. Създава подзаявки. | 6 |
| 9. Обяснява разликата между обединение и съединение на данни. | 8 |
| 10. Имплементира групиране на данни, както и филтрирането и сортирането им. | 6 |
| 11. Сравнява функциите, съхранените процедури и транзакциите. | 4 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 6 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира понятието бази данни и различава типовете данни. | **6** | 1 | 1 |  |  |
| 2. Демонстрира знанията си за създаване, промяна и премахване на таблици в бази данни. | **6** |  |  | 1 |  |
| 3. Дефинира процеса на нормализация и диференцира нормал- ните форми. | **10** | 1 |  |  | 1 |
| 4. Посочва, различава и демонстрира типовете връзки в релаци- онните бази данни. | **14** | 2 | 1 | 1 |  |
| 5. Прави изводи за различните каскадни операции. | **8** |  |  |  | 1 |
| 6. Познава, обяснява и дава пример за ограничители (constraints). | **16** | 2 | 3 |  |  |
| 7. Познава, разбира и различава SQL заявки за основните опе- рации (CRUD). | **16** | 1 | 2 | 1 |  |
| 8. Създава подзаявки. | **6** |  |  | 1 |  |
| 9. Обяснява разликата между обединение и съединение на данни. | **8** |  |  |  | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 6 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 10. Имплементира групиране на данни, както и филтрирането и сортирането им. | 6 |  |  | 1 |  |
| 11. Сравнява функциите, съхранените процедури и транзакци- ите. | 4 |  | 1 |  |  |
| **Общ брой задачи:** | **23** | **7** | **8** | **5** | **3** |
| **Общ брой точки:** | **100** | **14** | **32** | **30** | **24** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 7:* **Разработка на софтуер**

Трислойни модели Model-View-Controller и Model-View-ViewModel – графично представяне, на- чин на изпълнение. Рефакториране на код. Видове тествания на кода (компонентен, интеграцио- нен, регресионен и системен). Писане на SQL заявки чрез език за програмиране. Използване на системи за обектно-релационно свързване (ORM frameworks) – разлики в подходите Code First и Database First. Разработка на приложение с ORM framework. Принципи за правилно разделяне на компонентите на приложението по слоеве. Пакети, пакетни мениджъри и тяхната употреба.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Налична подходяща версия на СУБД от изучаваните. Например MSSQL, MySQL, PostgreSQL, Oracle Database и др. Задачи и фрагменти от код на програмен език от изучава- ните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 7** | **Максимален брой точки** |
| 1. Дефиниране на трислойни модели (MVC и MVVM). Представя ги графично и различава начините на изпълнението им. | 18 |
| 2. Обяснява и модифицира (рефакторира) готов примерен код. | 10 |
| 3. Описва и различава видовете тествания. | 6 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 7** | **Максимален брой точки** |
| 4. Сравнява Code First и Database First подхода и прави изводи за разликите между тях. Описва процеса на разработка, следвайки подходите. | 12 |
| 5. Разработва част от приложение, което да работи с база от данни с помощта на Code First или Database First. | 20 |
| 6. Демонстрира работа с ORM технология и диференцира SQL заявките от ORM подхода. | 20 |
| 7. Дефинира понятието пакет и пакетен мениджър. Демонстрира инсталирането на пакети. Открива и инсталира подходящи пакети за изпълнението на конк- ретна задача. | 14 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 7 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефиниране на трислойни модели (MVC и MVVM). Предс- тавя ги графично и различава начините на изпълнението им. | **18** | 1 | 2 |  | 1 |
| 2. Обяснява и модифицира (рефакторира) готов примерен код. | **10** |  | 1 | 1 |  |
| 3. Описва и различава видовете тествания. | **6** | 1 | 1 |  |  |
| 4. Сравнява Code First и Database First подхода и прави изводи за разликите между тях. Описва процеса на разработка, след- вайки подходите. | **12** |  | 1 |  | 1 |
| 5. Разработва част от приложение, което да работи с база от данни с помощта на Code First или Database First. | **20** | 2 | 1 | 2 |  |
| 6. Демонстрира работа с ORM технология и диференцира SQL заявките от ORM подхода. | **20** |  |  | 2 | 1 |
| 7. Дефинира понятието пакет и пакетен мениджър. Демонст- рира инсталирането на пакети. Открива и инсталира подхо- дящи пакети за изпълнението на конкретна задача. | **14** | 1 |  | 2 |  |
| **Общ брой задачи:** | **21** | **5** | **6** | **7** | **3** |
| **Общ брой точки:** | **100** | **10** | **24** | **42** | **24** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 8:* **Разработка на софтуер**

Техники за търсене на дефекти. Работа с дебъгер. Рефакториране на код. Следването на добри практики при създаване на код. Употреба на стилови ръководства ("style guides") и линтери ("linters"). Писане на компонентни тестове и добри практики при рефакториране чрез писане на тестове. Интегрирани среди за разработка (IDE). Разлика между IDE и текстов редактор. Техники за продуктивно използване на IDE и текстови редактори – допълнителни разширения и клавишни комбинации. Използване на инструменти в IDE за работа със сорс-контрол системи. Използване на базови инструменти за търсене и промяна на кода. Инструменти за намиране на всички упот- реби на фрагмент код, навигиране към дефиниция на функция.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Налична подходяща версия на СУБД от изучаваните. Например MSSQL, MySQL, PostgreSQL, Oracle Database и др. Задачи и фрагменти от код на програмен език от изучава- ните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 8** | **Максимален брой точки** |
| 1. Описва предназначението и работата с дебъгер. | 4 |
| 2. Коректно рефакторира и модифицира зле написан код, който нарушава доб- рите практики и стилови ръководства. | 12 |
| 3. Описва и демонстрира създаването на компонентни тестове на предварително предоставен код. | 8 |
| 4. Дефинира и различава понятията интегрирана среда за разработка (IDE) и тек- стов редактор. | 6 |
| 5. Описва и демонстрира инструменти и клавишни комбинации за работа с код (бързо навигиране в кода, намиране на срещания и замяна, извличане – extract на методи и класове и др.) в интегрирана среда за разработка и текстов редак- тор. | 16 |
| 6. Описва и дава пример за употребата на инструменти в интегрирана среда за разработка и текстови редактори за работа със сорс-контрол система. | 14 |
| 7. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 40 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 8 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Описва предназначението и работата с дебъгер. | **4** | 2 |  |  |  |
| 2. Коректно рефакторира и модифицира зле написан код, който нарушава добрите практики и стилови ръководства. | **12** |  |  | 2 |  |
| 3. Описва, обяснява и демонстрира създаването на компонен- тни тестове на предварително предоставен код. | **12** | 1 |  | 1 |  |
| 4. Дефинира и различава понятията интегрирана среда за разра- ботка (IDE) и текстов редактор. | **6** | 1 | 1 |  |  |
| 5. Описва и демонстрира инструменти и клавишни комбинации за работа с код (бързо навигиране в кода, намиране на среща- ния и замяна, извличане – extract на методи и класове и др.) в интегрирана среда за разработка и текстов редактор. | **14** |  | 1 | 2 |  |
| 6. Описва и дава пример за употребата на инструменти в интег- рирана среда за разработка и текстови редактори за работа със сорс-контрол система. | **12** | 3 | 2 |  |  |
| 7. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **40** |  |  |  | 5 |
| **Общ брой задачи:** | **21** | **7** | **4** | **5** | **5** |
| **Общ брой точки:** | **100** | **14** | **16** | **30** | **40** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 9:* **Разработка на софтуер**

Създаване на конзолен и графичен потребителски интерфейс за приложение (уеб, десктоп или мобилно). Компоненти в графичен потребителски интерфейс. Технологии (от изучаваните) за създаване на графичен потребителски интерфейс. UML и E/R диаграми. ORM системи.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm,*

*Eclipse и др. Налична подходяща версия на СУБД от изучаваните. Например MSSQL, MySQL, PostgreSQL, Oracle Database и др. Задачи и фрагменти от код на програмен език от изучава- ните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 9** | **Максимален брой точки** |
| 1. Описва и обяснява видове потребителски интерфейс. | 6 |
| 2. Различава графичен и конзолен интерфейс. | 4 |
| 3. Посочва компоненти на графичен интерфейс и обяснява предназначението им. | 24 |
| 4. Описва стъпките за създаване на потребителски интерфейс с изучавана техно- логия и ги демонстрира. | 10 |
| 5. Свързва потребителския интерфейс и бизнес-логиката на приложението. | 12 |
| 6. Разработва програмен код (класове, методи и др.) спрямо UML и E/R диаг- рами. Разчита UML и E/R диаграми. | 12 |
| 7. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 32 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 9 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Описва и обяснява видове потребителски интерфейс. | **6** | 1 | 1 |  |  |
| 2. Различава графичен и конзолен интерфейс. | **4** |  | 1 |  |  |
| 3. Посочва компоненти на графичен интерфейс и обяснява предназначението им. | **24** | 4 | 4 |  |  |
| 4. Описва стъпките за създаване на потребителски интерфейс с изучавана технология и ги демонстрира. | **10** | 2 |  | 1 |  |
| 5. Свързва потребителския интерфейс и бизнес-логиката на приложението. | **12** |  |  | 2 |  |
| 6. Разработва програмен код (класове, методи и др.) спрямо UML и E/R диаграми. Разчита UML и E/R диаграми. | **12** |  |  | 2 |  |
| 7. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **32** |  |  |  | 4 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 9 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **Общ брой задачи:** | **22** | **7** | **6** | **5** | **4** |
| **Общ брой точки:** | **100** | **14** | **24** | **30** | **32** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 10:* **Операционни системи**

Структура и ресурси на компютърната система (процесор, оперативна памет, периферни устройс- тва). Структура, основни функции и компоненти на операционна система. Архитектура на опе- рационните системи – монолитни, многослойни и архитектура с микроядра. Файлова структура. Пакетни системи. Процеси и памет. Виртуална памет. Услуги в ОС: ssh (keys), ftp, scp, мрежови услуги, dns, dhcp. Стартиране и спиране на услуги ръчно и по график. Файлови системи. Shell програмиране. Създаване и изпълнение на shell скриптове. Виртуализация и контейнери.

**Дидактически материали:** *Компютър с подходяща версия на Linux-базирана операционна сис- тема или подходящо софтуерно обезпечение (емулатор, виртуална машина), което поддържа изучаваните команди в Linux.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 10** | **Максимален брой точки** |
| 1. Посочва, различава и демонстрира знания за отделните хардуерни компо- ненти на компютърна система. | 12 |
| 2. Обяснява структурата на операционната система. Демонстрира знания за ар- хитектурата на операционните системи – монолитни, многослойни и архитек- тура с микроядра. | 10 |
| 3. Прави заключения и изводи за файловата структура на ОС, при конкретна пос- тавена задача. | 8 |
| 4. Посочва, обяснява и демонстрира команди, чрез които се показва функцио- налността на операционните системи. | 12 |
| 5. Обяснява пакетните системи в ОС. | 4 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 10** | **Максимален брой точки** |
| 6. Диференцира процесите в операционните системи. Демонстрира знания за виртуалната памет на конкретна операционна система. | 14 |
| 7. Изброява и обяснява услуги в ОС. При конкретна поставена задача за старти- ране и спиране на услуга ръчно или по график избира правилния начин. | 14 |
| 8. Прави заключения и изводи за файловите системи на различните операционни системи. | 8 |
| 9. Обяснява основни оператори в shell програмирането. Демонстрира знания за създаване на shell скриптове. | 10 |
| 10. Прави изводи за виртуализация и контейнери. | 8 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 10 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Посочва, различава и демонстрира знания за отделните хар- дуерни компоненти на компютърна система. | **12** | 1 | 1 | 1 |  |
| 2. Обяснява структурата на операционната система. Демонст- рира знания за архитектурата на операционните системи – монолитни, многослойни и архитектура с микроядра. | **10** |  | 1 | 1 |  |
| 3. Прави заключения и изводи за файловата структура на ОС, при конкретна поставена задача. | **8** |  |  |  | 1 |
| 4. Посочва, обяснява и демонстрира команди, чрез които се по- казва функционалността на операционните системи. | **12** | 1 | 1 | 1 |  |
| 5. Обяснява пакетните системи в ОС. | **4** |  | 1 |  |  |
| 6. Диференцира процесите в операционните системи. Демонст- рира знания за виртуалната памет на конкретна операционна система. | **14** |  |  | 1 | 1 |
| 7. Изброява и обяснява услуги в ОС. При конкретна поставена задача за стартиране и спиране на услуга ръчно или по график избира правилния начин. | **14** | 1 | 1 |  | 1 |
| 8. Прави заключения и изводи за файловите системи на различ- ните операционни системи. | **8** |  |  |  | 1 |
| 9. Обяснява основни оператори в shell програмирането. Демон- стрира знания за създаване на shell скриптове. | **10** |  | 1 | 1 |  |
| 10. Прави изводи за виртуализация и контейнери. | **8** |  |  |  | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 10 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **Общ брой задачи:** | **19** | **3** | **6** | **5** | **5** |
| **Общ брой точки:** | **100** | **6** | **24** | **30** | **40** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 11:* **Програмиране за вградени системи**

Основи на електротехниката и електрониката. Блокова схема на вградена система. Видове раз- войни платки. Микроконтролери. Електронни елементи и материали в електрониката. Елект- ронни схеми и принципи на действие. Модулации. Широчинно-импулсна модулация. Среда за разработка (IDE). Структура на програма за вградена система. Типове данни. Константи, промен- ливи и оператори. Управление състоянието на изводите (pins). Вградени функции. Управление на електронни компоненти с развойни платки. Прост цифров и аналогов вход. Управление с PWM на произволен цифров изход. Серийна комуникация.

**Дидактически материали:** *Компютър с подходяща версия на интегрирана среда за разработка и/или емулатор за микроконтролер от изучаваните. Например Arduino IDE, Tinkercad и др. За- дачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 11** | **Максимален брой точки** |
| 1. Дефинира понятията: електричество, променлив ток (AC), прав ток (DC), ши- рочинно-импулсна модулация и др. | 4 |
| 2. Различава видовете развойни платки и микроконтролери. | 4 |
| 3. Обяснява основните характеристики и особености на вградените системи. Дава пример за блокова схема на вградена система. | 8 |
| 4. Обяснява и дава пример за принципа на работа на вградена система. Избира електронни компоненти за вградена система по поставена задача. | 16 |
| 5. Разработва и проектира вградена система, така че да реши поставена задача. | 6 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 11** | **Максимален брой точки** |
| 6. Описва специфичните изисквания и особености на софтуера, предназначен за вградени системи. | 2 |
| 7. Обяснява структурата на програма за вградена система. | 4 |
| 8. Демонстрира код за управление на състоянието на изводите. | 6 |
| 9. Демонстрира код за управление на електронни компоненти с развойна платка по зададена задача. | 6 |
| 10. Прави заключения и изводи за серийната комуникация. | 8 |
| 11. Открива грешки в програмен код и го модифицира, така че да реши поставе- ната задача. | 12 |
| 12. Анализира, определя и допълва програмен код, така че да реши поставената задача. | 24 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 11 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира понятията: електричество, променлив ток (AC), прав ток (DC), широчинно-импулсна модулация и др. | **4** | 2 |  |  |  |
| 2. Различава видовете развойни платки и микроконтролери. | **4** |  | 1 |  |  |
| 3. Обяснява основните характеристики и особености на вграде- ните системи. Дава пример за блокова схема на вградена сис- тема. | **8** |  | 2 |  |  |
| 4. Обяснява и дава пример за принципа на работа на вградена система. Избира електронни компоненти за вградена система по поставена задача. | **16** |  | 2 |  | 1 |
| 5. Разработва и проектира вградена система, така че да реши поставена задача. | **6** |  |  | 1 |  |
| 6. Описва специфичните изисквания и особености на софтуера, предназначен за вградени системи. | **2** | 1 |  |  |  |
| 7. Обяснява структурата на програма за вградена система. | **4** |  | 1 |  |  |
| 8. Демонстрира код за управление на състоянието на изводите. | **6** |  |  | 1 |  |
| 9. Демонстрира код за управление на електронни компоненти с развойна платка по зададена задача. | **6** |  |  | 1 |  |
| 10. Прави заключения и изводи за серийната комуникация. | **8** |  |  |  | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 11 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 11. Открива грешки в програмен код и го модифицира, така че да реши поставената задача. | **12** |  |  | 2 |  |
| 12. Анализира, определя и допълва програмен код, така че да реши поставената задача. | **24** |  |  |  | 3 |
| **Общ брой задачи:** | **19** | **3** | **6** | **5** | **5** |
| **Общ брой точки:** | **100** | **6** | **24** | **30** | **40** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 12:* **Конкурентно програмиране**

Конкурентност. Изпълнение на програма. Процес. Блокиращи операции. Видове блокиращи опе- рации. Нишка. Връзка между процес и нишка. Създаване на нишки. Управление на нишки. Син- хронизация между нишки. Проблеми при работа с нишки и асинхронни операции. Работа с асин- хронни операции. Обещания (Promise/Task) и обратни извиквания (Callback). Работа с асинх- ронни операции чрез async/await и други механизми за реализиране на асинхронни операции.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 12** | **Максимален брой точки** |
| 1. Дефинира понятията: конкурентност, процес, блокираща операция, нишка. | 8 |
| 2. Дава пример за блокиращи операции. | 4 |
| 3. Прави изводи за проблемите и решенията им при работа с нишки и асинх- ронни операции. | 16 |
| 4. Описва връзката между процес и нишка. Сравнява процес и нишка. | 6 |
| 5. Демонстрира създаване и управление на нишки. | 12 |
| 6. Обобщава начините за синхронизацията между нишки. | 8 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 12** | **Максимален брой точки** |
| 7. Обяснява начина на работа на асинхронните операции. | 4 |
| 8. Дефинира понятията обещание (Promise/task) и обратно извикване (callback) (или аналогичните им) в изучаван език за програмиране. | 4 |
| 9. Демонстрира асинхронни операции. | 6 |
| 10. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 32 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 12 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Дефинира понятията: конкурентност, процес, блокираща операция, нишка. | **8** | 4 |  |  |  |
| 2. Дава пример за блокиращи операции. | **4** |  | 1 |  |  |
| 3. Прави изводи за проблемите и решенията им при работа с нишки и асинхронни операции. | **16** |  |  |  | 2 |
| 4. Описва връзката между процес и нишка. Сравнява процес и нишка. | **6** | 1 | 1 |  |  |
| 5. Демонстрира създаване и управление на нишки. | **12** |  |  | 2 |  |
| 6. Обобщава начините за синхронизацията между нишки. | **8** |  | 2 |  |  |
| 7. Обяснява начина на работа на асинхронните операции. | **4** |  | 1 |  |  |
| 8. Дефинира понятията обещание (Promise/task) и обратно из- викване (callback) (или аналогичните им) в изучаван език за програмиране. | **4** | 2 |  |  |  |
| 9. Демонстрира асинхронни операции. | **6** |  |  | 1 |  |
| 10. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **32** |  |  |  | 4 |
| **Общ брой задачи:** | **21** | **7** | **5** | **3** | **6** |
| **Общ брой точки:** | **100** | **14** | **20** | **18** | **48** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 12 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 13:* **Математически основи на програмирането**

Бройни системи, операции и преобразувания. Генерална съвкупност и извадка. Средна стойност, мода и медиана. Графични представяния на статистически данни – полигон, хистограма, кръгова диаграма. Свойства на функциите. Правоъгълна координатна система. Изобразяване на графика на функция. Системи линейни уравнения – методи за решаване. Вектор – свойства, връзка между вектори и масиви в програмирането. Множества. Операции с множества. Комбинаторика. Ос- новни комбинаторни конфигурации – пермутации, комбинации и вариации. Елементи от теория на вероятностите. Събития, вероятност на събитие, условна вероятност. Пресмятане на вероят- ности. Създаване и/или поправка/допълване на вече съществуващи компютърни програми със средствата на програмен език.

**Дидактически материали:** *Компютър с подходяща среда за интегрирана разработка и/или подходяща среда за компютърна математика от изучаваните. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 13** | **Максимален брой точки** |
| 1. Посочва бройни системи. Преобразува числа от една бройна система в друга. Изчислява изрази с числа в различна бройна система. | 12 |
| 2. Дефинира понятия от статистиката – генерална съвкупност и извадка, средна стойност, мода и медиана. Представя графично статистически данни. | 12 |
| 3. Демонстрира решаването на системи линейни уравнения с повече неизвестни с помощта на компютър. | 6 |
| 4. Дефинира понятието функция. Посочва свойства на функциите. Демонстрира изчертаване на графики на математически функции с помощта на компютър. | 8 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 13** | **Максимален брой точки** |
| 5. Дефинира понятието вектор. Посочва свойства на векторите. Обяснява връз- ката между векторите и масивите в програмирането. | 8 |
| 6. Дефинира понятието множество. Посочва операции с множества. Представя графично операции с множества. | 12 |
| 7. Посочва видовете комбинаторни конфигурации. Различава пермутации, ком- бинации и вариации. Демонстрира алгоритми за генериране на комбинаторни конфигурации. | 12 |
| 8. Демонстрира пресмятане на вероятности с помощта на компютър. | 6 |
| 9. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 24 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 13 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Посочва бройни системи. Преобразува числа от една бройна система в друга. Изчислява изрази с числа в различна бройна система. | **12** | 1 | 1 | 1 |  |
| 2. Дефинира понятия от статистиката – генерална съвкупност и извадка, средна стойност, мода и медиана. Представя гра- фично статистически данни. | **12** | 2 |  |  | 1 |
| 3. Демонстрира решаването на системи линейни уравнения с повече неизвестни с помощта на компютър. | **6** |  |  | 1 |  |
| 4. Дефинира понятието функция. Посочва свойства на функци- ите. Демонстрира изчертаване на графики на математически функции с помощта на компютър. | **8** | 1 |  | 1 |  |
| 5. Дефинира понятието вектор. Посочва свойства на векторите. Обяснява връзката между векторите и масивите в програми- рането. | **8** | 2 | 1 |  |  |
| 6. Дефинира понятието множество. Посочва операции с мно- жества. Представя графично операции с множества. | **12** | 2 |  |  | 1 |
| 7. Посочва видовете комбинаторни конфигурации. Различава пермутации, комбинации и вариации. Демонстрира алго- ритми за генериране на комбинаторни конфигурации. | **12** | 1 | 1 | 1 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 13 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 8. Демонстрира пресмятане на вероятности с помощта на ком- пютър. | **6** |  |  | 1 |  |
| 9. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **24** |  |  |  | 3 |
| **Общ брой задачи:** | **22** | **9** | **3** | **5** | **5** |
| **Общ брой точки:** | **100** | **18** | **12** | **30** | **40** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 14:* **Интернет програмиране**

Мрежови протоколи (IP, TCP, UDP). Видове HTTP заявки (GET/POST/PUT/DELETE/PATCH). Клиент-сървърна комуникация. Основни тагове в HTML. Селектори и основни правила в CSS. Създаване на семантични страници. Създаване на адаптивно (responsive) оформление на стра- ници. Увод в JavaScript – работа с обекти и събития. Принципи и манипулиране на DOM

**Дидактически материали:** *Компютър с подходяща версия на интегрирана среда за разработка от изучаваните и уеб браузър. Например Visual Studio, Visual Studio Code, SublimeText, Atom и др. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 14** | **Максимален брой точки** |
| 1. Обяснява предназначението и диференцира различните протоколи. | 12 |
| 2. Дефинира понятието HTTP заявка, прави изводи за различните HTTP методи и избира метод за конкретна ситуация. | 18 |
| 3. Обяснява и представя графично клиент-сървърната комуникация. | 12 |
| 4. Различава смисъла на употребата и необходимостта от HTML, CSS и JavaScript. | 8 |
| 5. Дефинира и използва коректно HTML тагове. | 8 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 14** | **Максимален брой точки** |
| 6. Задава свойства на HTML компонентите чрез CSS. | 6 |
| 7. Описва и демонстрира употребата на семантични елементи за създаване на семантична страница. | 8 |
| 8. Обяснява и демонстрира начините за създаване на адаптивен (responsive) ди- зайн. | 10 |
| 9. Работа с обекти и събития в JavaScript. Ученикът пише код и модифицира съ- ществуващ такъв спрямо конкретна задача. | 12 |
| 10. Демонстрира манипулирането на DOM. | 6 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 14 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Обяснява предназначението и диференцира различните про- токоли. | **12** |  | 1 |  | 1 |
| 2. Дефинира понятието HTTP заявка, прави изводи за различ- ните HTTP методи и избира метод за конкретна ситуация. | **18** | 1 |  |  | 2 |
| 3. Обяснява и представя графично клиент-сървърната комуни- кация. | **12** |  | 1 |  | 1 |
| 4. Различава смисъла на употребата и необходимостта от HTML, CSS и JavaScript. | **8** |  |  |  | 1 |
| 5. Дефинира и използва коректно HTML тагове. | **8** | 1 |  | 1 |  |
| 6. Задава свойства на HTML компонентите чрез CSS. | **6** |  |  | 1 |  |
| 7. Описва и демонстрира употребата на семантични елементи за създаване на семантична страница. | **8** | 1 |  | 1 |  |
| 8. Обяснява и демонстрира начините за създаване на адаптивен (responsive) дизайн. | **10** |  | 1 | 1 |  |
| 9. Работа с обекти и събития в JavaScript. Ученикът пише код и модифицира съществуващ такъв спрямо конкретна задача. | **12** |  |  | 2 |  |
| 10. Демонстрира манипулирането на DOM. | **6** |  |  | 1 |  |
| **Общ брой задачи:** | **18** | **3** | **3** | **7** | **5** |
| **Общ брой точки:** | **100** | **6** | **12** | **42** | **40** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 14 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 15:* **Интернет програмиране**

MVC модел. Бази данни и ORM технологии. Реализиране на CRUD операции. Създаване на шаб- лонни изгледи. Упълномощаване (authentication) и удостоверяване (authorization). Управление на сесии и бисквитки. Често срещани уязвимости в сигурността на уеб приложенията (SQL инжек- ция, XSS атака, CRSF, LFI и RFI, DDoS атака, MITM атака). Принципи на REST API. Работа с REST API в JSON/XML формат. Използване на AJAX в REST API.

**Дидактически материали:** *Компютър с подходяща версия на интегрирана среда за разработка от изучаваните, уеб браузър и сървърен софтуер. Наличие на подходяща версия на СУБД от изучаваните. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 15** | **Максимален брой точки** |
| 1. Обяснява и представя графично MVC модела. | 12 |
| 2. Описва същността и демонстрира употребата на ORM технологиите. | 8 |
| 3. Реализира CRUD операции. | 24 |
| 4. Дефинира и създава шаблонен изглед. | 8 |
| 5. Различава автентикацията и авторизацията. | 8 |
| 6. Диференцира сесиите от бисквитките. | 8 |
| 7. Обяснява и дава пример за различните уязвимости в сигурността на уеб при- ложенията. Прави изводи за предотвратяването им. | 16 |
| 8. Обяснява основните принципи на REST API. | 4 |
| 9. Възпроизвежда и обяснява код за работа с REST API в JSON/XML формат. | 6 |
| 10. Възпроизвежда и обяснява код за използване на AJAX в REST API. | 6 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 15 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Обяснява и представя графично MVC модела. | **12** |  | 1 |  | 1 |
| 2. Описва същността и демонстрира употребата на ORM техно- логиите. | **8** | 1 |  | 1 |  |
| 3. Реализира CRUD операции. | **24** |  |  | 4 |  |
| 4. Дефинира и създава шаблонен изглед. | **8** | 1 |  | 1 |  |
| 5. Различава автентикацията и авторизацията. | **8** |  |  |  | 1 |
| 6. Диференцира сесиите от бисквитките. | **8** |  |  |  | 1 |
| 7. Обяснява и дава пример за различните уязвимости в сигур- ността на уеб приложенията. Прави изводи за предотвратява- нето им. | **16** |  | 2 |  | 1 |
| 8. Обяснява основните принципи на REST API. | **4** |  | 1 |  |  |
| 9. Възпроизвежда и обяснява код за работа с REST API в JSON/XML формат. | **6** | 1 | 1 |  |  |
| 10. Възпроизвежда и обяснява код за използване на AJAX в REST API. | **6** | 1 | 1 |  |  |
| **Общ брой задачи:** | **20** | **4** | **6** | **6** | **4** |
| **Общ брой точки:** | **100** | **8** | **24** | **36** | **32** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 16:* **Софтуерно инженерство**

Етапи в разработката на софтуер. Софтуерни изисквания и документация. Случаи на употреба (use cases) и потребителски истории (user stories). Спецификация на изискванията. Методологии за разработка на софтуер – Waterfall и Agile. Scrum. Scrum артефакти, екипи, роли и събития. Инструменти за управление на екип (Jira или други изучавани). Работа със системи за сорс-кон- трол (git, svn или друга изучавана). Работа с чужд код. Преглед на чужд код (code review). Видове софтуерно тестване.

**Дидактически материали:** *Компютър с подходяща версия на интегрирана среда за разработка от изучаваните Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 16** | **Максимален брой точки** |
| 1. Описва и обяснява етапите в разработката на софтуер. | 6 |
| 2. Посочва и различава методологии за разработка на софтуер. Прави заключе- ния и изводи за значението на методологии за разработка на софтуер. | 14 |
| 3. Свързва SCRUM артефакти, събития и роли. Обобщава и прави изводи за SCRUM артефакти, събития и роли. | 18 |
| 4. Описва и обобщава съдържанието на софтуерната документация. | 20 |
| 5. Различава случаи на употреба (use cases) и потребителски истории (user stories). | 4 |
| 6. Познава и демонстрира инструменти за управление на работата на екип. | 8 |
| 7. Демонстрира команди за работа със система за сорс-контрол. | 12 |
| 8. Демонстрира процеса на преглед на чужд код (code review). | 4 |
| 9. Описва и различава видове софтуерно тестване. Обобщава и диференцира употребата на различните видове софтуерно тестване. | 14 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 16 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Описва и обяснява етапите в разработката на софтуер. | **6** | 1 | 1 |  |  |
| 2. Посочва и различава методологии за разработка на софтуер. Прави заключения и изводи за значението на методологии за разработка на софтуер. | **14** | 1 | 1 |  | 1 |
| 3. Обяснява и свързва SCRUM артефакти, събития и роли. Обобщава и прави изводи за SCRUM артефакти, събития и роли. | **18** |  | 1 | 1 | 1 |
| 4. Описва и обобщава съдържанието на софтуерната докумен- тация. Прави изводи и заключения за значението на софтуер- ната документация в работата на програмистите. | **20** | 2 | 2 |  | 1 |
| 5. Различава случаи на употреба (use cases) и потребителски ис- тории (user stories). | **4** |  | 1 |  |  |
| 6. Познава и демонстрира инструменти за управление на рабо- тата на екип. | **8** | 1 |  | 1 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 16 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 7. Демонстрира команди за работа със система за сорс-контрол. | **12** |  |  | 2 |  |
| 8. Обобщава процеса на преглед на чужд код (code review). | **4** |  | 1 |  |  |
| 9. Описва и различава видове софтуерно тестване. Обобщава и диференцира употребата на различните видове софтуерно тестване. | **14** | 1 | 1 |  | 1 |
| **Общ брой задачи:** | **22** | **6** | **8** | **4** | **4** |
| **Общ брой точки:** | **100** | **12** | **32** | **24** | **32** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 17:* **Функционално програмиране**

Функционална парадигма на програмиране. Понятие за „странични ефекти“ в програмирането. Входно/изходни операции. „Състояние“ на програмата. Функции и рекурсия. Чисти функции. Работа със списъци: „глава“ и „опашка“.

**Дидактически материали:** *Компютър с подходяща версия на интегрирана среда за разра- ботка. Задачи и фрагменти от код на програмен език от изучаваните.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 17** | **Максимален брой точки** |
| 1. Описва функционалната парадигма на програмиране. Различава функцио- нална, процедурна и обектно-ориентирана парадигма. Прави изводи за значе- нието на функционалната парадигма на програмиране. | 14 |
| 2. Дефинира понятието „страничен ефект“ в програмирането. | 2 |
| 3. Посочва и демонстрира начини за извършване на входно/изходни операции чрез функционален език за програмиране. | 14 |
| 4. Обяснява понятието „състояние“ на програма. | 4 |
| 5. Описва и демонстрира създаване на функции. | 10 |
| 6. Разработва рекурсивни функции чрез функционален език за програмиране. | 6 |
| 7. Дефинира понятието „чиста функция“. | 2 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 17** | **Максимален брой точки** |
| 8. Дава пример и обобщава обработката на списъци с функции. | 16 |
| 9. Различава глава и опашка на списък. | 8 |
| 10. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 24 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 17 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Описва функционалната парадигма на програмиране. Разли- чава функционална, процедурна и обектно-ориентирана па- радигма. Прави изводи за значението на функционалната па- радигма на програмиране. | **14** | 1 | 1 |  | 1 |
| 2. Дефинира понятието „страничен ефект“ в програмирането. | **2** | 1 |  |  |  |
| 3. Посочва и демонстрира начини за извършване на входно/из- ходни операции чрез функционален език за програмиране. | **14** | 1 |  | 2 |  |
| 4. Обяснява понятието „състояние“ на програма. | **4** |  | 1 |  |  |
| 5. Описва и демонстрира създаване на функции. | **10** | 2 |  | 1 |  |
| 6. Разработва рекурсивни функции чрез функционален език за програмиране. | **6** |  |  | 1 |  |
| 7. Дефинира понятието „чиста функция“. | **2** | 1 |  |  |  |
| 8. Дава пример и обобщава обработката на списъци с функции. | **16** |  | 2 |  | 1 |
| 9. Различава глава и опашка на списък. | **8** |  | 2 |  |  |
| 10. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **24** |  |  |  | 3 |
| **Общ брой задачи:** | **21** | **6** | **6** | **4** | **5** |
| **Общ брой точки:** | **100** | **12** | **24** | **24** | **40** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 17 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

*Изпитна тема № 18:* **Алгоритми и структури от данни**

Алчни алгоритми – понятие, примери, решаване на задачи с алчни алгоритми. Рекурсия и рекур- сивни алгоритми – понятие, примери за рекурсивни алгоритми и алгоритми по схемата на пъл- ното изчерпване (backtracking). Комбинаторни съединения и алгоритми за генерирането им: ва- риации, комбинации, пермутации – понятия, алгороитми, прилики и разлики. Динамично опти- миране – техника, едномерни и двумерни задачи. Теория на графите – основни понятия, предс- тавяне на графи в програмен код. Алгоритми върху графи – алгоритми за обхождане на граф, графично представяне на алгоритмите и имплементация с програмен код.

**Дидактически материали:** *Компютър с подходяща версия на софтуерното обезпечение и ин- тегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др.*

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 18** | **Максимален брой точки** |
| 1. Описва понятието алчен алгоритъм. Дава пример за алчни алгоритми. | 6 |
| 2. Описва понятието рекурсия. Дава пример за рекурсивни алгоритми с пълно изчерпване. | 6 |
| 3. Разработва програми за решаването на задачи с помощта на алчни алгоритми. | 6 |
| 4. Дефинира понятията от комбинаториката: вариации, комбинации и пермута- ции. Различава вариации, комбинации и пермутации. | 6 |
| 5. Обяснява техниката на динамичното оптимиране. | 4 |
| 6. Разработва програми за решението на едномерни и двумерни задачи с по- мощта на техники от динамичното оптимиране (задача за раницата, най-малка подредица, оптимален път в лабиринт и др.) | 6 |

|  |  |
| --- | --- |
| **Критерии за оценяване на изпитна тема № 18** | **Максимален брой точки** |
| 7. Дефинира основни понятия от теория на графите: граф, ориентиран и неори- ентиран граф, връх, ребро, път в граф, цикъл в граф. | 14 |
| 8. Обяснява и представя графично алгоритми за обхождане в ширина и дълбо- чина на графи (BFS и DFS). | 20 |
| 9. Анализира фрагмент/и от код и идентифицира и поправя правилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необходимо. | 32 |
| **ОБЩ БРОЙ ТОЧКИ:** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 18 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| 1. Описва понятието алчен алгоритъм. Дава пример за алчни ал- горитми. | **6** | 1 | 1 |  |  |
| 2. Описва понятието рекурсия. Дава пример за рекурсивни ал- горитми с пълно изчерпване. | **6** | 1 | 1 |  |  |
| 3. Разработва програми за решаването на задачи с помощта на алчни алгоритми. | **6** |  |  | 1 |  |
| 4. Дефинира понятията от комбинаториката: вариации, комби- нации и пермутации. Различава вариации, комбинации и пер- мутации. | **6** | 1 | 1 |  |  |
| 5. Обяснява техниката на динамичното оптимиране. | **4** |  | 1 |  |  |
| 6. Разработва програми за решението на едномерни и двумерни задачи с помощта на техники от динамичното оптимиране (задача за раницата, най-малка подредица, оптимален път в лабиринт и др.) | **6** |  |  | 1 |  |
| 7. Дефинира основни понятия от теория на графите: граф, ори- ентиран и неориентиран граф, връх, ребро, път в граф, цикъл в граф. Разработва програми с представяне на графи. | **14** | 1 |  | 2 |  |
| 8. Обяснява и представя графично алгоритми за обхождане в ширина и дълбочина на графи (BFS и DFS). | **20** |  | 1 |  | 2 |
| 9. Анализира фрагмент/и от код и идентифицира и поправя пра- вилно грешките в написания програмен код, така че да реши поставената задача. Допълва кода, ако и когато това е необ- ходимо. | **32** |  |  |  | 4 |
| **Общ брой задачи:** | **19** | **4** | **5** | **4** | **6** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Матрица на писмен тест по изпитна тема № 18 Критерии за оценяване** | **Максимален брой точки** | **Брой тестови задачи по равнища** | | | |
| **I** | **II** | **III** | **IV** |
| **Знание 0 – 2 т.** | **Разбиране 0 – 4 т.** | **Приложение 0 – 6 т.** | **Анализ 0 – 8 т.** |
| *1* | *2* | *3* | *4* | *5* | *6* |
| **Общ брой точки:** | **100** | **8** | **20** | **24** | **48** |
| **При оценка на резултатите от теста максимален брой точки се поставя при отговор, съвпадащ с ключа за оценяване, както следва:**   * **2 точки за тестовите задачи от равнище „Знание“;** * **4 точки за тестовите задачи от равнище „Разбиране“;** * **6 точки за тестовите задачи от равнище „Приложение“;** * **8 точки за тестовите задачи от равнище „Анализ“.** | | | | | |

## Критерии и показатели за оценка на дипломния проект и неговата защита

*(Попълва се индивидуално от председателя и членовете на комисията)*

|  |  |  |
| --- | --- | --- |
| **Критерии и показатели за оценяване** | **Максимален брой точки за показателите** | **Максимален брой точки за критерия** |
| **1. Съответствие с изискванията за съдържание и**  **структура на дипломния проект** |  | **20** |
| 1.1. логическа последователност и структура на изло-  жението, балансиране на отделните части | 4 |  |
| 1.2. задълбоченост и пълнота при формулиране на  обекта, предмета, целта и задачите в разработването на темата | 7 |  |
| 1.3. използване на подходящи изследователски ме-  тоди | 4 |  |
| 1.4. стил и оформяне на дипломната работа (термино- логия, стил на писане, текстообработка и оформяне на фигури и таблици, качество на програмния код и спаз- ване на добри практики за разработка на софтуер, ако  проектът съдържа софтуерна част) | 5 |  |
| **2. Съответствие между поставените цели на дип-**  **ломния проект и получените резултати** |  | **20** |
| 2.1. изводите следват пряко от изложението, форму- лирани са ясно, решават поставените в началото на из- следването цели и задачи и водят до убедителна за-  щита на поставената теза | 10 |  |
| 2.2. оригиналност, значимост и актуалност на темата | 6 |  |
| 2.3. задълбоченост и обоснованост на предложенията  и насоките | 4 |  |
| **3. Представяне на дипломния проект** |  | **20** |

|  |  |  |
| --- | --- | --- |
| **Критерии и показатели за оценяване** | **Максимален брой точки за**  **показателите** | **Максимален брой точки за**  **критерия** |
| 3.1. представянето на разработката по темата e ясно и  точно | 5 |  |
| 3.2. онагледяване на експозето с:  а) презентация;  б) графични материали;  в) демонстрация на софтуерен проект, ако е изготвен такъв;  г) практически резултати;  д) компютърна мултимедийна симулация и анимация. | 10 |  |
| 3.3. умения за презентиране | 5 |  |
| **4. Отговори на зададените въпроси от рецензента**  **и/или членовете на комисията за защита на дипло- мен проект** |  | **30** |
| 4.1. разбира същността на зададените въпроси и отго-  варя пълно, точно и убедително | 10 |  |
| 4.2. логически построени и точни отговори на зададе-  ните въпроси | 10 |  |
| 4.3. съдържателни и обосновани отговори на въпро-  сите | 10 |  |
| **5. Използване на професионалната терминология,**  **добър и ясен стил, обща езикова грамотност** |  | **10** |
| 5.1. правилно използване на професионалната терми-  нология | 5 |  |
| 5.2. ясен изказ и обща езикова грамотност | 5 |  |
| **Общ брой точки:** | **Максимален**  **бр. точки 100** | **Максимален**  **бр. точки 100** |

## ІV. ДЪРЖАВЕН ИЗПИТ ЗА ПРИДОБИВАНЕ НА ПРОФЕСИОНАЛНА КВАЛИФИ- КАЦИЯ – ЧАСТ ПО ПРАКТИКА НА ПРОФЕСИЯТА

1. **Указание за съдържанието на индивидуалните задания**

Индивидуалното задание по практика съдържа темата на индивидуалното задание и изиск- ванията към крайния резултат от изпълнението на заданието. По решение на комисията за про- веждане и оценяване на изпита – част по практика на професията, могат да се дадат допълнителни указания, които да подпомогнат обучавания при изпълнение на индивидуалното задание.

Индивидуалното задание се състои в създаване на приложен софтуер, който задължително включва следните три компонента:

* + потребителски интерфейс
  + система за управление на бази от данни
  + част, базирана на многослоен модел, която координира потребителския интерфейс и сис- темата за управление на бази от данни

Приложението може да бъде уеб, десктоп или мобилно-базирано приложение.

## Примерно индивидуално практическо задание № 1

**Тема: РАЗРАБОТВАНЕ НА МНОГОСЛОЙНО ПРИЛОЖЕНИЕ ЗА УПРАВЛЕНИЕ НА ПОРЪЧКИ НА ХРАНА ЗА ДОМА**

## Указания (инструкции/изисквания) за изпълнение на индивидуалното задание:

* Да се изберат подходяща архитектура на приложението и структура на базата данни спрямо конкретната задача.
* Да се създаде слой за модела на приложението, включващ всички бизнес обекти и процеси.
* Да се обособи функционалността, свързана с основните операции (*CRUD*) за боравене с ба- зата от данни.
* Да се избере подходящ презентационен слой (*графичен, мобилен, уеб*).
* Да се добави възможност за упълномощаване (*authentication*) и удостоверяване (*authorization*) на потребителите, използващи програмния продукт.
* Да има функционалност, която филтрира и групира данните по определени критерии.
* Да има функционалност, която позволява прикрепянето на изображения.
* Да има функционалност, която позволява записването и работата с дати (*например изчислява разлика между две дати в брой дни и др.*)
* Да се спазват добрите практики и основни принципи за писане на качествен програмен код.

## Критерии за оценяване

|  |  |  |
| --- | --- | --- |
| **Критерии и показатели за оценяване** | **Максимален брой точки** | **Тежест** |
| **1. Спазване на правилата за здравословни и безопасни условия на труд и опазване на околната среда** | | **да/не** |
| 1.1. Изпълнява дейностите при спазване на необходимите мерки за оси- гуряване на здравословни и безопасни условия на труд |  |  |
| 1.2. Създава организация за осигуряване на здравословни и безопасни условия на труд на работното място |  |  |
| 1.3. Предотвратява опасните ситуации, които могат да възникнат по време на работа |  |  |

* Критерий 1 няма количествено изражение, а качествено. Ако обучаваният по време на изпита създава опасна ситуация, застрашаваща собствения му живот или живота на други лица, изпитът се прекратява и на обуча- вания се поставя **оценка слаб (2).**

|  |  |  |
| --- | --- | --- |
| **Критерии и показатели за оценяване** | **Максимален брой точки** | **Тежест** |
| **2. Ефективна организация на работното място** | | **5** |
| 2.1. Планира ефективно работния процес | 2 |  |
| 2.2. Разпределя трудовите дейности в работния процес съобразно пос- тавената задача и времето за нейното изпълнение | 2 |
| 2.3. Познава и прилага установените стандарти за осъществяване на дейността | 1 |
| **3. Спазване изискванията на правилниците, наредбите и предписанията** | | **5** |
| 3.1. Познава и прилага нормативните изисквания в съответната профе- сионална област | 3 |  |
| 3.2. Спазва изискванията на правилниците, наредбите и предписанията, свързани с индивидуалното задание | 2 |
| **4. Правилен подбор на технологични софтуерни и хардуерни средства, детайли, материали и инструменти съобразно конкретното задание** | | **20** |
| 4.1. Целесъобразно използва софтуерни и хардуерни средства, матери- али, детайли и инструменти според заданието | 10 |  |
| 4.2. Правилно подбира, организира и конфигурира софтуерните и хар- дуерните средства. Преценя правилно количеството и качеството на ма- териали, детайли и инструменти, ако са необходими | 10 |
| **5. Спазване на технологичната последователност на операциите и добрите прак- тики според индивидуалното задание** | | **20** |
| 5.1. Самостоятелно определя технологичната последователност на опе- рациите. Създава адекватна софтуерна архитектура на проекта (прило- жимо за задания, свързани със създаване на приложен софтуер) | 10 |  |
| 5.2. Организира дейността си при спазване на технологичната последо- вателност на операциите в процеса на работа. Съобразява се с устано- вената софтуерна архитектура на проекта (приложимо за задания, свър- зани със създаване на приложен софтуер) | 5 |
| 5.3. Спазва добрите практики при изпълнение на заданието – създаване на четим и ясен код, а в по-комплексни ситуации, детайлно описание, посредством коментари, създава документация, ако това е необходимо | 5 |  |
| **6. Качество на изпълнението на индивидуалното задание** | | **50** |
| 6.1. Всяка завършена дейност съответства на изискванията на съответ- ната технология, които се посочват в заданието | 20 |  |
| 6.2. Крайният резултат съответства на зададените параметри и отговаря на изискванията в заданието | 20 |
| 6.3. Изпълнява задачата в поставения срок | 10 |

|  |  |  |
| --- | --- | --- |
| **Критерии и показатели за оценяване** | **Максимален брой точки** | **Тежест** |
| **Общ брой точки:** | **100** | **100** |

## СИСТЕМА ЗА ОЦЕНЯВАНЕ

Оценяването на резултатите от държавния изпит за придобиване на трета степен на профе- сионална квалификация по **специалността код 4810301 „Приложно програмиране“, професия код 481030 „Приложен програмист“** е в точки, както следва:

* част по теория на професията – максимален брой 100 точки;
* част по практика на професията – максимален брой 100 точки.

Всяка част от държавния изпит е успешно положена при постигане на петдесет на сто от максималния брой точки.

Формирането на окончателната оценка от изпита е в съотношение – 50 процента от получе- ния брой точки от частта по теория на професията и 50 процента от получения брой точки от частта по практика на професията.

Окончателната оценка в брой точки се формира след успешното полагане на всяка част от изпита и се изчислява, както следва:

Окончателната оценка в брой точки е равна на 0,5 × получения брой точки от частта по теория на професията + 0,5 × получения брой точки от частта по практика на професията.

Окончателната оценка от брой точки се превръща в цифрова оценка с точност до 0,01 по формулата: **Цифрова оценка = окончателната оценка в брой точки х 0,06.**

Окончателната оценката от държавния изпит за придобиване на квалификация по професи- ята е с количествен и качествен показател, с точност до 0,01 и се определя, както следва:

а) за количествен показател от 2,00 до 2,99 се определя качествен показател „слаб“; б) за количествен показател от 3,00 до 3,49 се определя качествен показател „среден“; в) за количествен показател от 3,50 до 4,49 се определя качествен показател „добър“;

г) за количествен показател от 4,50 до 5,49 се определя качествен показател „много добър“; д) за количествен показател от 5,50 до 6,00 се определя качествен показател „отличен“.

## ПРЕПОРЪЧИТЕЛНА ЛИТЕРАТУРА

* 1. Азълов, П. Обектно-ориентирано програмиране. Структури от данни и STL. Сиела, 2008. ISBN: 978-954-28-0184-9.
  2. Баркър, К., Photoshop задкулисни трикове за дизайнери, 2017, ISBN: 9789546563330.
  3. Василев, Ал., Програмиране с Java™ SE 10 – основи на езика в примери, 2018, ISBN: 978-619-7356-53-3.
  4. Генков, Д., Основи на компютърните мрежи, 2014, ISBN: 978-619-7071-61-0.
  5. Дюис, Гл. Инструментите на Photoshop, 2018, ISBN: 9789546563569.
  6. Енков, Св. Програмиране в среда Arduino. Пловдив, 2017, ISBN: 978-619-202-261-7.
  7. Иванов, М. Софтуерни приложения в среда Android, 2016, ISBN: 9789545359170.
  8. Илиева С., Лилов В., Манова И. Подходи и методи за реализация на софтуерни системи, УИ „Св. Климент Охридски“, 2010, ISBN: 9789540729992.
  9. Каракехайов, Здр., Кнуд Кристенсен и Оле Винтер. Проектиране на вградени микроком- пютърни системи с микроконтролери. Хардуер и софтуер, София – Москва, 2000, ISBN: 954-642-101-4.
  10. Колектив. Компютърни мрежи, СофтПрес, 2005, ISBN: 9789546853691.
  11. Колисниченко, Д. Програмиране на мобилни приложения за Windows 10, Асеневци, 2017, ISBN: 9786197356069.
  12. Колисниченко Д. HTML 5 & CSS 3 – практическо програмиране за начинаещи. Трето издание, Асеневци, София, 2017, ISBN 9786197356113.
  13. Колисниченко, Д. SQL практическо програмиране. Асеневци, 2018, ISBN: 9786197356434.
  14. Комолова, Н. и Яковлева, Е. CorelDRAW X8, 2017, ISBN: 9786197356205.
  15. Лонг Д., Java в облака. Spring Boot, Spring Cloud и Cloud Foundry, Асеневци, София, 2019, ISBN 9786197356649.
  16. Макмилън, Тр., Cisco: Компютърни мрежи – основи, АлексСофт, 2016, ISBN: 9789546563156.
  17. Наков С. и колектив. Принципи на програмирането със C#. Фабер. 2018. ISBN 978-619- 00-0778-4. https://introprogramming.info/intro-csharp-book/.
  18. Наков С. и колектив. Основи на програмирането със C#. Фабер. 2017. ISBN: 978-619-00- 0635-0. https://csharp-book.softuni.bg/.
  19. Патиас, Йо., Васил Георгиев. Проектиране на роботизирани системи. София, 2017, ISBN: 9789540742076.
  20. Пенева, Ю. Принципи на базите от данни. Нов български университет, 2018, ISBN: 9789545359996.
  21. Петков, Д. Микроконтролери: Архитектура и принцип на действие. София, 2015.
  22. Петров Р. Микропроцесорни системи и микроконтролери. София, 2011, ISBN: 9789542907046.
  23. Петрова – Антонова Д. Проектиране и интегриране на софтуерни системи с .NET тех- нологии. УИ „Св. Климент Охридски“, 2012, ISBN: 9789540733012.
  24. Рашидов Алд. Инсталиране и конфигуриране на Web сървъри под Linux и Windows, 2012, ISBN: 9789544902988.
  25. Стоянов, Б. Ръководство за упражнения по програмиране на асемблер. Шумен, 2011, ISBN: 978-954-9775-37-2.
  26. Стоянова З. Здравословни и безопасни условия на труд за 9. клас., БГ Учебник, 2021.
  27. Терзиева Т., Създаване на графичен потребителски интерфейс на C#. Практическо ръ- ководство. Университетско издателство „Паисий Хилендарски“, 2015, ISBN: 978- 6192020392.
  28. Тодоров, Ас., Валентин Моллов и Кирил Мечков. Ръководство за лабораторни упраж- нения по микропроцесорна техника. ТУ – София, 2014, ISBN: 978-619-167-128-1.
  29. Тодорова, М. Обектно-ориентирано програмиране на базата на езика С++. Сиела. 2011. ISBN: 978-954-28-0909-8.
  30. Толева-Стоименова, С., С. Бойчева. Дискретна математика. Теоретични основи на ин- форматиката. Сиела, София, 2018, ISBN 978-954-28-2743-6.
  31. Улман Л., PHP7 за Web. Бързо ръководство. АлексСофт, София, 2017, ISBN 9789546563408.
  32. Шиндър, Д. Компютърни мрежи. СофтПрес, 2008, ISBN: 9546852540.
  33. Academy, D. K. Компютърни мрежи. Наръчник на системния администратор. Асеневци, 2018, ISBN: 9786197356502.
  34. Academy, D. K. Практическо програмиране за Android TM. Асеневци, 2018, ISBN: 9786197356557.
  35. Голощапов, Ал. Google Android: Программирование для мобильных устройств. Санкт Петербург, БХВ-Петербург, 2012, ISBN: 978-5-9775-0729-5.
  36. Соммер, У. Программирование микроконтроллерных плат Arduino/Freeduino (+CD), БХВ-Петербург, 2012, ISBN: 978-5-9775-0727-1.
  37. Attaway S., MATLAB: A Practical Introduction to Programming and Problem Solving 5th Edition, Butterworth-Heinemann, 2018, ISBN 978-0128154793.
  38. Bauer C. Java Persistence with Hibernate, Manning Publications, 2015, ISBN: 978- 1617290459.
  39. Blain, John M. The Complete Guide to Blender Graphics, Second Edition: Computer Modeling and Animation, 2015, ISBN-13: 978-1482216639.
  40. Downey A., Think Stats, O’Reilly, 2011, ISBN 9781449307110, достъпна в електронен вид: https://greenteapress.com/thinkstats/html/index.html.
  41. Farrell P. Math Adventures with Python, No Starch Press, San Francisco, 2019, ISBN 1593278675.
  42. Feathers M. Working Effectively with Legacy Code, Pearson, 2004, ISBN: 9780131177055.
  43. Freeman A. Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor.
  44. Galloway J. Professional ASP.NET MVC 5, ISBN: 978-1118794753.
  45. Iversen, J., Eierman, M. Learning Mobile App Development: A Hands-on Guide to Building Apps with iOS and Android, London, 2014, ISBN-13: 978-0-321-94786-4.
  46. Jason van Gumster, Blender For Dummies, 2020, ISBN-13: 978-1119616962.
  47. Loeliger J. & Mccullough M., Version Control with Git: Powerful Tools And Techniques For Collaborative Software Development Second Edition, O'Reilly Media, 2012, ISBN: 9781449316389.
  48. Margolis, Michael. Arduino Cookbook, O'Reilly Media, 2011, ISBN: 0596802471,978- 0596802479.
  49. Martin C. R. Clean Code: A Handbook of Agile Software Craftsmanship, Pearson, 2008, ISBN: 9780132350884.
  50. МcDowell G. Cracking the Coding Interview: 189 Programming Questions and Solutions, CarrerCup, 2015, ISBN: 0984782850.
  51. McGrath M. Linux in easy steps, 6th edition – illustrated using Linux Mint, In Easy Steps Limited, 2018, ISBN 978-1840788082.
  52. McKinney W. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O’Reilly, 2017, ISBN 978-1491957660.
  53. Monk, S. Programming Arduino: Getting Started with Sketches, Second Edition, McGraw- Hill Education, 2016, ISBN: 978-1259641633.
  54. Murach J. Murach's ASP.NET Core MVC, Apress, 2020, ISBN: 978-1-4842-3149-4.
  55. Loeliger J. & Mccullough M., Version Control with Git: Powerful Tools And Techniques For Collaborative Software Development Second Edition, O'Reilly Media, 2012, ISBN: 9781449316389.

56. Pages, Apress, 2020, ISBN: 978-1-4842-5439-4.

1. Poulton N., Docker Deep Dive, Independently Published, 2017, ISBN 978-1521822807.
2. Pressman R., Software Engineering: A Practitioner‘s Approach, McGraw-Hill, 2000, ISBN: 0073655783.
3. Robbins A. Bash Pocket Reference, O'Reilly Media, 2016, ISBN 978-1491941591.
4. Silberschatz A., Gagne G., Galvin P. Operating Systems Concepts, Wiley, 2012, ISBN 978- 1118063330.
5. Smith J. Entity Framework Core in Action, Manning Publications, 2018, ISBN: 978- 1617294563.
6. Tanenbaum А. Modern Operating Systems (4th Edition), Pearson, 2014, ISBN 978- 0133591620.
7. Woodhull A. Operating Systems: Design and Implementation, 3rd edition, Pearson, 2006, ISBN 978-0136373315.

Електронни информационни източници:

1. Портал за еOбучение в специалност „Приложен програмист“ https://it-kariera.mon.bg/e-learning (посетен 02.09.2021 г.).
2. Свободно учебно съдържание по програмиране и ИТ за българските училища на Работна група „Образование по програмиране и ИТ“

https://github.com/BG-IT-Edu (посетен 02.09.2021 г.).

## VIІ. АВТОРСКИ ЕКИП

1. Доц. д-р Димитър Петров Минчев – Бургаски свободен университет
2. Инж. Петър Росенов Петров – Професионална гимназия по електротехника и електроника „Константин Фотинов“, Бургас
3. Инж. Хриси Атанасова Плачкова – Математическа гимназия „Акад. Кирил Попов“, Пловдив
4. Галя Колева Митева – Професионална гимназия по електротехника и електроника

„Константин Фотинов“, Бургас

1. Иван Илиев Илиев – Математическа гимназия „Акад. Кирил Попов“, Пловдив

## VІІІ. ПРИЛОЖЕНИЯ

1. **Изпитен билет – част по теория на професията**

………………………………………………………………………............................................

*(пълно наименование на училището/обучаващата институция)*

# ДЪРЖАВЕН ИЗПИТ

**ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВА- ЛИФИКАЦИЯ – ЧАСТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА**

**по професия код 481030 „Приложен програмист“ специалност код 4810301 „Приложно програмиране“**

# Изпитен билет № 1

## Изпитна тема: ИЗЧИСЛЕНИЯ, ЛИНЕЙНИ, РАЗКЛОНЕНИ И ЦИКЛИЧНИ АЛГОРИТМИ

Дефиниция на основни понятия: програмиране, език за програмиране, алгоритъм, среда за разработка (IDE), компилация и интерпретация. Променливи и типове от данни – дефини- ция, сравнение на типове и употреба. Пресмятания, аритметични и логически оператори, из- рази – видове и разработка. Описание на условни конструкции – пълна и кратка форма, вло- жени условни конструкции. Логически изрази и оператори за сравнение – видове и разработка. Оператор за многовариантен избор – описание, сравнение с условен оператор. Циклични опе- ратори – описание и разработка. Видове цикли – сравнение, прилики и разлики. Вложени цикли

– приложение. Същност на подпрограми (функции/методи), видове параметри, видове връ- щана стойност, предимства при употребата на подпрограми. Определяне на реда на изпълнение на фрагментите в кода (program flow). Работа със среда за разработка (IDE) – създаване и за- реждане на проект в среда за разработка. Създаване и/или поправка/допълване на вече същес- твуващи компютърни програми, решаващи изчислителни и алгоритмични задачи със средст- вата на програмен език.

Разполагате с фрагмент на C#, който е свързан с посочената по-долу задача. Анализирайте фрагмента и го поправете, променете и/или допълнете, така че да получите работеща програма, която решава поставената по-долу задача. Програмен фрагмент:

int n = int.Parse(Console.ReadLine);

for(int i = 0; i < n; i-) {

int number == int.Parse(Console.NewLine()) sum = number;

if(number > 0) { positiveSum + number; positiveCount++;

}

if(number % 3 == 0) countDivisibleBy3++

}

}

Console.WriteLine(sum);

int positiveAverage = positiveSum / positiveCount; Console.WriteLine(positiveAverage); Console.WriteLine(countDivisibleBy3);

**Задача 1.** Да се състави програма, която въвежда цяло число n и последователност от n на брой цели числа, всяко на отделен ред. Програмата да намира:

* + сумата от всички числа;
  + средно-аритметично от положителните числа;
  + броя на числата, които са кратни на 3.

Примерен вход и изход:

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 5  3  8  - 3  7  6 | 21  6  3 |

**Описание на дидактическите материали:** *Компютър с подходяща версия на софтуерното обез- печение и интегрирана среда за разработка от изучаваните. Например Visual Studio, IntelliJ IDEA, PyCharm, Eclipse и др. Задачи и фрагменти от код на програмен език от изучаваните.*

**Председател на изпитната комисия**:........................................................................... .....................

*(име, фамилия) (подпис)*

**Директор/ръководител на обучаващата институция:**............................................. ……………

*(име, фамилия) (подпис) (печат на училището/обучаващата институция)*

## Индивидуално задание по практика

………………………………………………………………………............................................

*(пълно наименование на училището/обучаващата институция)*

## ДЪРЖАВЕН ИЗПИТ

**ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ**

## – ЧАСТ ПО ПРАКТИКА НА ПРОФЕСИЯТА

**по професия код 481030 „Приложен програмист“ специалност код 4810301 „Приложно програмиране“**

## И н д и в и д у а л н о з а д а н и е № 1

На ученика/обучавания ........................................................................................................................

*(трите имена на ученика/обучавания)*

от .................клас/курс, начална дата на изпита: …….……………....……. начален час: ..............

крайна дата на изпита: .............................................. час на приключване на изпита:…...................

## Да се разработи многослойно приложение за управление на поръчки на храна за дома.

1. **Указания (инструкции/изисквания) за изпълнение на практическото задание:**

* Да се изберат подходяща архитектура на приложението и структура на базата данни спрямо конкретната задача.
* Да се създаде слой за модела на приложението, включващ всички бизнес обекти и процеси.
* Да се обособи функционалността, свързана с основните операции (CRUD) за боравене с базата от данни.
* Да се избере подходящ презентационен слой (графичен, мобилен, уеб).
* Да се добави възможност за упълномощаване (authentication) и удостоверяване (authorization) на потребителите, използващи програмния продукт.
* Да има функционалност, която филтрира и групира данните по определени критерии.
* Да има функционалност, която позволява прикрепянето на изображения.
* Да има функционалност, която позволява записването и работата с дати (например изчис- лява разлика между две дати в брой дни и др.).
* Да се спазват добрите практики и основни принципи за писане на качествен програмен код.

**УЧЕНИК/ОБУЧАВАН:** ………………………………………………...……….. ...........................

*(име, фамилия) (подпис)*

**Председател на изпитната комисия:**.................................................................... ..........................

*(име, фамилия) (подпис)*

**Директор/ръководител на обучаващата институция:**........................................ ……………….

*(име, фамилия) (подпис) (печат на училището/обучаващата институция)*

## Указание за разработване на писмен тест

При провеждане на държавния изпит – част теория на професията, с писмен тест въз основа на критериите за оценка към всяка изпитна тема се съставят тестовите задачи.

Всяка тестова задача задължително съдържа поне един глагол (при възможност започва с глагол), изразяващ действието, което трябва да извърши обучаваният, и показващ равнището по таксономията на Блум, еталона на верния отговор и ключ за оценяване – пълния отговор, за който се получават максимален брой точки съобразно равнището на задачата, определени в таблицата за критериите за оценка на всяка изпитна тема.

Към всеки тест се разработва указание за работа, което включва целта на теста – какви зна- ния и умения се оценяват с него; описание на теста – брой задачи, типология (задачи със свободен отговор; задачи за допълване/съотнасяне; задачи с изборен отговор) и начин на работа с тях; про- дължителност на времето за работа с теста; начин на оценяване на резултатите от теста.

## А) Примерно указание за работа

………………………………………………………………………....................................................................

*(пълно наименование на училището/обучаващата институция)*

## УВАЖАЕМИ УЧЕНИЦИ/КУРСИСТИ,

Вие получавате тест, който съдържа …. задачи с различна трудност с максимален брой точки – 100. За всеки Ваш отговор ще получите определен брой точки, показан в долния десен ъгъл след всяка задача.

Целта на теста е да се установи равнището на усвоените от Вас знания и умения, задъл- жителни за придобиване на трета степен на професионална квалификация по **професия „При- ложен програмист“**, **специалност „Приложно програмиране“**.

Отбелязването на верния според Вас отговор при задачите с изборен отговор е чрез знак

**X**, а за другите типове задачи начинът на отговор е описан в задачата.

**Запомнете!** Като действителен отговор на съответната задача се приема само този, отбе- лязан със знака **X**.

Някои задачи изискват не само познаване на учебното съдържание, но и логическо мис- лене, затова четете внимателно условията на задачите, преди да посочите някой отговор за ве- рен.

Не отделяйте много време на въпрос, който Ви се струва труден, върнете се на него по- късно, ако Ви остане време.

Тестът е с продължителност астрономически часа.

## ПОЖЕЛАВАМЕ ВИ УСПЕШНА РАБОТА!

**Б) Методически указания за комисията за подготовка и оценяване на изпита – част теория на професията, за разработването и оценяването на писмения тест:**

Броят и равнището на тестовите задачи по всеки критерий се определят съобразно равни- щето, на което трябва да бъде усвоено съответното учебно съдържание, като общият брой задачи по всеки критерий трябва да носи максималния брой точки.

## а) Таксономия на Блум – равнища и примерни глаголи

|  |  |  |
| --- | --- | --- |
| **Равнище** | **Характеристика** | **Глаголи** |
| **І. Знание**  **0 – 2 точки** | Възпроизвеждане и разпознаване на информация за понятия, факти, дефи- ниции | Дефинира, описва, посочва, избро- ява, очертава, възпроизвежда, фор- мулира, схематизира |
| **ІІ. Разбиране 0 – 4 точки** | Извличане на съществен смисъл от изучаваната материя. Интерпретация и трансформиране на информацията с цел нейното структуриране | Преобразува, различава, обяснява, обобщава, преразказва, решава, дава пример за…, сравнява |
| **ІІІ. Приложение 0 – 6 точки** | Пренос на нови знания и умения при решаване на проблемна или аварийна ситуация. Способност за използване на усвоената информация и формираните умения | Изчислява, демонстрира, открива, модифицира, разработва, свързва, доказва |
| **ІV. Анализ** | Разкриване на взаимовръзки, зависи- мости, тенденции и формулиране на изводи и заключения | Разделя, подразделя, диферен- цира, различава, представя гра- фично, определя, илюстрира, прави заключения и изводи, обоб- щава, избира, разделя, подразделя |

**б) Препоръчителни тестови въпроси и задачи според типа на отговора I група: въпроси и задачи със свободен отговор**

* Въпроси и задачи за свободно съчинение
* Въпроси и задачи за тълкуване

## група: въпроси и задачи за допълване (с полуоткрит отговор)

* Въпроси и задачи за допълване на дума или фраза, или елемент от чертеж/схема
* Въпроси и задачи за заместване

## група: въпроси и задачи с изборен отговор

* Задачи с един или повече верни отговори
* Въпроси за избор между вярно и грешно

## В) Примерни тестови задачи

**ПРИМЕРНА ТЕСТОВА ЗАДАЧА ОТ РАВНИЩЕ „ЗНАНИЕ“**

Посочете кой от следните модификатори на достъп позволява достъп до член на класа само от него или от неговите наследници:

а) public

б) protected в) internal г) private

**Макс. 2 т.**

## Еталон на верния отговор: б)

**Ключ за оценяване:**

Отговор б) – 2 точки

При посочени повече от един отговор – 0 точки Всички останали отговори – 0 точки

**ПРИМЕРНА ТЕСТОВА ЗАДАЧА ОТ РАВНИЩЕ „РАЗБИРАНЕ“**

Попълнете таблицата, която определя нивото на достъп за различните модификатори на достъп в C#, така че всички колони да са правилни:

|  |  |
| --- | --- |
| **Модификатор** | **Достъп** |
|  | Навсякъде в решението (solution) |
|  | Навсякъде в асемблито (assembly) |
| private |  |
| protected |  |

## Еталон на верния отговор и ключ за оценяване:

**Макс. 4 т.**

|  |  |
| --- | --- |
| **Модификатор** | **Достъп** |
| public – 1 т. | Навсякъде в решението (solution) |
| internal – 1 т. | Навсякъде в асемблито (assembly) |
| private | Само в класа, в който е дефиниран – 1 т. |
| protected | В класа, който е дефиниран, и всички кла- сове, които го наследяват – 1т. |

**ПРИМЕРНА ТЕСТОВА ЗАДАЧА ОТ РАВНИЩЕ „ПРИЛОЖЕНИЕ“**

Предоставен е клас BankAccount.cs. Допълнете програмния фрагмент със следните методи:

a) void Withdraw(decimal amount), който изтегля определено количество пари от сметката; б) void Deposit(decimal amount), който добавя определено количество пари към сметката.

class BankAccount

{

public string IBAN { get; private set; } public decimal Balance { get; private set; }

public BankAccount(string iban, decimal balance)

{

IBAN = iban; Balance = balance;

}

public override string ToString()

{

return $"IBAN: {IBAN}; Balance: {Balance:F2}";

}

}

## Еталон на верния отговор: а)

**Макс. 6 т.**

public void Withdraw(decimal amount)

{

if(amount > Balance)

{

throw new InvalidOperationException("The amount to be withdrawn should not exceed the balance.");

}

Balance -= amount;

}

***Забележка:*** *Може да се приеме за коректна и друга имплементация на метода, стига функци-*

*оналността да работи коректно.*

## б)

public void Deposit(decimal amount)

{

if(amount <= 0)

{

throw new InvalidOperationException("The amount to be deposited should not be negaitve");

}

Balance += amount;

}

***Забележка:*** *Може да се приеме за коректна и друга имплементация на метода, стига функци- оналността да работи коректно.*

## Ключ за оценяване:

При коректна реализация на метод а) – 3 точки При коректна реализация на метод б) – 3 точки

При частична реализация на метод а) или б) – до 2 точки, в зависимост от допуснатата грешка При некоректна или липсваща реализация на метод – 0 точки

**ПРИМЕРНА ТЕСТОВА ЗАДАЧА ОТ РАВНИЩЕ „АНАЛИЗ“**

Открийте и поправете допуснатите синтактични и логически грешки в следния програмен фраг- мент. Допълнете кода, ако това е нужно:

class Person

{

// Име

public string name; public string Name

{

get

{

return name;

set

}

if (value.Length < 3)

{

throw new ArgumentException("Name cannot be less than 3

symbols");

}

name = value;

}

}

private int age; public int Age

{

get

{

return age;

}

set

{

**Макс. 8 т.**

if (value <= 0)

{

throw new ArgumentException("Age cannot be zero or

negative integer");

}

Age = value;

}

}

public Person(string name, int age)

{

this.Name = name; this.Age = age;

}

}

## Еталон на верния отговор:

Програмен фрагмент с нанесени корекции:

class Person

{

// Име

**private** string name; public string Name

{

get

{

return name;

**}**

set

**{**

if (value.Length < 3)

{

throw new ArgumentException("Name cannot be less than 3

symbols");

}

name = value;

}

}

private int age; public int Age

{

get

{

return age;

}

set

{

if (value <= 0)

{

throw new ArgumentException("Age cannot be zero or

negative integer");

}

**а**ge = value;

}

}

public Person(string name, int age)

{

this.Name = name; this.Age = age;

}

}

## Ключ за оценяване:

За всяка правилно посочена и отстранена грешки се дават по 2 т.

## Индивидуално задание за разработване на дипломен проект

……………………………………………………………………….....................................................

*(пълно наименование на училището)*

## ЗАДАНИЕ ЗА ДИПЛОМЕН ПРОЕКТ

**ДЪРЖАВЕН ИЗПИТ ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИО- НАЛНА КВАЛИФИКАЦИЯ – ЧАСТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА**

## по професия код 481030 „Приложен програмист“ специалност код 4810301 „Приложно програмиране“

На .................................................................................................... ............................. ученик/ученичка от клас

*(трите имена на ученика)*

**Тема: РАЗРАБОТКА НА УЕБ САЙТ НА СТРОИТЕЛНА ФИРМА „МЕЧТАН ДОМ“**

**Изисквания за разработката на дипломния проект (входни данни, съдържание, оформяне, указания за из- пълнение, инструкции):**

Фирма „Мечтан дом“ се занимава със строителство на жилищни и офис сгради. Фирмата е построила десетки обекти и се нуждае от уеб сайт, в който да може да се публикуват всички проекти на фирмата.

Сайтът трябва да предлага възможност за разглеждане на наличните апартаменти, магазини, паркоместа и други обекти в дадена сграда и да съдържа информация за статуса на даден имот (наличен/продаден/резервиран).

Сайтът трябва да разполага със следните публично достъпни секции: „Начало“, „За нас“,

„Оферти“, „Новини“, „Контакти“.

Сайтът трябва да може да се администрира с помощта на администраторско меню, което да позволи добавяне на нови оферти или промяна на някоя от другите страници.

Дизайнът на сайта трябва да е адаптивен и да изглежда еднакво добре на различни устройства. Изберете аргументирано подходящи технологии.

**График за изпълнение:**

**а) дата на възлагане на дипломния проект ………………………………….…….**

**б) контролни проверки и консултации …………………………………………….**

**………………………………………**

**……………………………………….**

**……………………………………… в) краен срок за предаване на дипломния проект …………………………..……**

**Ученик**: …………………………………………………………………………………….………….. ...........................

*(име, фамилия) (подпис)*

**Ръководител-консултант**:...................................................................................................................... ..... .....................

*(име, фамилия) (подпис)*

**Директор**:................................................................................................................................. ……………………………

*(име, фамилия) (подпис)*

*(печат на училището)*

## Указания за съдържанието и оформянето на дипломния проект

**А. Съдържание на дипломния проект:**

Оформяне на дипломния проект в следните структурни единици:

* титулна страница;
* съдържание;
* увод (въведение);
* основна част;
* заключение;
* списък на използваната литература;
* приложения.

**Титулната страница** съдържа наименование на училището, населено място, тема на дипломния проект, трите имена на ученика, професия и специалност, име и фамилия на ръководителя/кон- султанта.

**Уводът** (*въведение*) съдържа кратко описание на основните цели и резултати.

**Основна част** – Формулира се целта на дипломния проект и задачите, които трябва да бъдат решени, за да се постигне тази цел. Съдържа описание и анализ на известните решения, като се цитират съответните литературни източници. Съдържа приносите на дипломния проект, които трябва да бъдат така формулирани, че да се вижда кои от поставените задачи са успешно решени. **Заключението** съдържа изводи и предложения за доразвиване на проекта и възможностите за неговото приложение.

**Списъкът с използваната литература** включва цитираната и използвана в записката на дип- ломния проект литература. Започва на отделна страница от основния текст. При имената на ав- торите първо се изписва фамилията. Всички описания в списъка с използваните източници трябва да са подредени по азбучен ред според фамилията на първия автор на всяка публикация.

**Приложенията** съдържат документация, която не е намерила място в текста поради ограничени- ята в обема й или за по-добра прегледност подредба. В текста трябва да има препратка към всички приложения.

## Б. Оформяне на дипломния проект

Формат: А4; Брой редове в стр.: 30; Брой на знаците: 60 знака в ред Общ брой на знаците в 1 стр.: 1800 – 2000 знака

Шрифт: Times New Roman

## Рецензия на дипломен проект

**WWWWWWWWWW1234567890123456789012345678901234567890123456789012**

*(пълно наименование на училището)*

# РЕЦЕНЗИЯ

|  |  |
| --- | --- |
| **Тема на дипломния проект** |  |
| **Ученик** |  |
| **Клас** |  |
| **Професия** |  |
| **Специалност** |  |
| **Ръководител-консултант** |  |
| **Рецензент** |  |

|  |  |  |
| --- | --- | --- |
| **Критерии за допускане до защита на дипломен проект** | **Да** | **Не** |
| Съответствие на съдържанието и точките от заданието |  |  |
| Съответствие между тема и съдържание |  |  |
| Спазване на препоръчителния обем на дипломния проект |  |  |
| Спазване на изискванията за оформление на дипломния проект |  |  |
| Готовност за защита на дипломния проект |  |  |

|  |  |
| --- | --- |
| Силни страни на дипломния проект |  |
| Допуснати основни слабости |  |
| Въпроси и препоръки към дипломния проект |  |

## З А К Л Ю Ч Е Н И Е

Качествата на дипломния проект дават основание ученикът/ученичката……………………………

……………………………….............................. да бъде допуснат/а до защита пред членовете на комисията за подготовка, провеждане и оценяване на изпит чрез защита на дипломен проект – част по теория на професията.

………..20... г. Рецензент: ……………………………

Гр./с……………. *(име и фамилия)*