Автономная некоммерческая образовательная организация
высшего образования
«Воронежский экономико-правовой институт»
(АНОО ВО «ВЭПИ»)

АНОО ВО "ВОРОНЕЖСКИЙ ЭКОНОМИКО-ПРАВОВОЙ ИНСТИТУТ", АНОО ВО "ВЭПИ", ВЭПИ

Подписано цифровой подписью: АНОО ВО "ВОРОНЕЖСКИЙ ЭКОНОМИКО-ПРАВОВОЙ ИНСТИТУТ", АНОО ВО "ВЭПИ", ВЭПИ
Дата: 26.09.2023 11:59:44 +03'00'

УТВЕРЖДАЮ
Проректор
по учебно-методической работе
_____ А.Ю. Жильников
«___» _____ 2023 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Б1.В.11 Объектно-ориентированное программирование
(наименование дисциплины (модуля))

09.03.03 Прикладная информатика
(код и наименование направления подготовки)

Направленность (профиль) Прикладная информатика в экономике
(наименование направленности (профиля))

Квалификация выпускника Бакалавр
(наименование квалификации)

Форма обучения Очная, заочная
(очная, заочная)

Рекомендована к использованию Филиалами АНОО ВО «ВЭПИ»

Воронеж 2023

Фонд оценочных средств по дисциплине (модулю) рассмотрен и одобрен на заседании кафедры прикладной информатики.

Протокол от «01» сентября 2023 г. № 1

Фонд оценочных средств по дисциплине (модулю) согласован со следующими представителями работодателей или их объединений, направление деятельности которых соответствует области профессиональной деятельности, к которой готовятся обучающиеся:

директор Ле /Попов Р.Н./ ООО "Ангелы Айти"
(должность, наименование организации, фамилия, инициалы, подпись, дата, печать)

директор /Кабанов Е.В./. ООО "Сфера Айти"
(должность, наименование организации, фамилия, инициалы, подпись, дата, печать)

Заведующий кафедрой М.С. Агафонова

Разработчики:

Доцент А.И. Кустов

1. Перечень компетенций с указанием этапов их формирования в процессе освоения ОП ВО

Целью проведения дисциплины Б1.В.11 Объектно-ориентированное программирование является достижение следующих результатов обучения:

Код компетенции	Наименование компетенции
ПК-2	способен разрабатывать и адаптировать прикладное программное обеспечение.
ПК-4	способен составлять технико-экономическое обоснование проектных решений и техническое задание на разработку информационной системы.

В формировании данных компетенций также участвуют следующие дисциплины (модули), практики образовательной программы (по семестрам (курсам) их изучения):

- для очной формы обучения:

Наименование дисциплин (модулей), практик	Этапы формирования компетенций по семестрам изучения							
	1 сем.	2 сем.	3 сем.	4 сем.	5 сем.	6 сем.	7 сем.	8 сем.
Исследование операций и методы оптимизации					ПК-4	ПК-4		
Технологическое предпринимательство							ПК-4	
Разработка приложений на платформе 1С					ПК-2	ПК-2	ПК-2	ПК-2
Разработка информационных систем на языках высокого уровня						ПК-2	ПК-2	
Технические измерения и приборы		ПК-2						
Имитационное моделирование в экономике					ПК-4			
Производственная практика (технологическая (проектно-технологическая) практика)						ПК-2		
Производственная практика (эксплуатационная практика)								ПК-4
Производственная практика (преддипломная практика)								ПК-2; ПК-4
Подготовка к процедуре защиты и защита выпускной квалификационной работы								ПК-2; ПК-4

- для заочной формы обучения:

Наименование дисциплин (модулей), практик	Этапы формирования компетенций по курсам изучения				
	1 курс	2 курс	3 курс	4 курс	5 курс
Исследование операций и методы оптимизации			ПК-4		
Технологическое предпринимательство					ПК-4
Разработка приложений на платформе 1С			ПК-2	ПК-2	
Разработка информационных систем на языках высокого уровня				ПК-2	ПК-2
Технические измерения и приборы		ПК-2			
Имитационное моделирование в экономике				ПК-4	
Производственная практика (технологическая (проектно-технологическая) практика)				ПК-2	
Производственная практика (эксплуатационная практика)					ПК-4
Производственная практика (преддипломная практика)					ПК-2; ПК-4
Подготовка к процедуре защиты и защита выпускной квалификационной работы					ПК-2; ПК-4

Этап дисциплины (модуля) Б1.В.11 Объектно-ориентированное программирование в формировании компетенций соответствует:

- для очной формы обучения – 3 семестру;
- для заочной формы обучения – 3 курсу.

2. Показатели и критерии оценивания компетенций на различных этапах их формирования, шкалы оценивания

Показателями оценивания компетенций являются следующие результаты обучения:

Код компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине (модулю)
ПК-2. Способен разрабатывать и адаптировать прикладное программное обеспечение.	ИПК-2.1. Применяет современные технологии разработки и адаптации прикладного программного обеспечения.	знать: - современные технологии разработки и адаптации прикладного программного обеспечения; уметь: - применять современные объектно-ориентированной технологии программирования; владеть: - навыками ввода и вывода данных в текстовом редакторе, электронной таблице и СУБД.
	ИПК-2.2. Участвует в разработке на современных языках программирования и адаптации компонентов прикладного программного обеспечения	знать: - общие сведения и особенности языка программирования офисных приложений VisualBasicforApplication (VBA) уметь: - применять современные языки программирования в разработки информационных продуктов; владеть: - навыками прикладного программного обеспечения.
ПК-4. Способен составлять технико-экономическое обоснование проектных решений и техническое задание на разработку информационной системы.	ИПК-4.1. Использует методики технико-экономического обоснования проектных решений.	знать: - основы информационного менеджмента, информационные технологии реализации проектных решений; уметь: - использовать методы разработки специализированных модулей и процедур в офисных программах; владеть: - навыками использования методики технико-экономического обоснования проектных решений.
	ИПК-4.2. Составляет техническое задание на разработку информационной системы.	знать: - основы разработки информационных систем; уметь: - составлять техническое задание на разработку информационной системы; владеть: - навыками обоснования проектных решений при разработке информационной системы или внедрения информационной технологии.
	ИПК-4.3. Участвует в исследовании эффективности функционирования информационных систем организации.	знать: - методы оценки эффективности функционирования информационных систем организации;

		уметь: - проводить оценку эффективности функционирования информационных систем организации; владеть: - современными информационными технологиями для экономического обоснования проектного решения.

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины (модуля):

№ п/п	Наименование раздела дисциплины (модуля)	Код компетенции, код индикатора достижения компетенции	Критерии оценивания	Оценочные средства текущего контроля успеваемости	Шкала оценивания
1	Тема 1. Основные концепции программирования в пакете офисных программ	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - сущность и содержание объектно-ориентированной технологии программирования Уметь: - программировать в VBA Владеть: - понятиями программного модуля	Опрос	«Зачтено» «Не зачтено»
2	Тема 2. Введение в макросы, создание макросов, работа с макрорекордером	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - понятие макроса и его интерпретация в VBA. Уметь: - разрабатывать макрос с помощью макрорекордера Владеть: - понятиями программного кода	Опрос	«Зачтено» «Не зачтено»
3	Тема 3. Реализация ввода и вывода данных в электронной таблице, текстовом редакторе и в СУБД	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - основные отличия в разработке программного кода специализированных процедур и функций в программах офисного пакета Уметь: - решать задачи Владеть: - навыками исследования возможностей по вводу и выводу данных в текстовом редакторе	Доклад	«Зачтено» «Не зачтено»
4	Тема 4. Разработка программ реализующих выполнение арифметических операций	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - алгоритм программных процедур Уметь: - исследовать возможности кодирования и отладки программных процедур Владеть: - навыками исследования	Опрос	

			возможности модификации программы для выполнения арифметических операций в текстовом редакторе		
5	Тема 5. Разработка графического интерфейса программы	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - понятие графического интерфейса Уметь: - применять основные инструменты при разработке формы в VBA Владеть: - навыками исследования возможности разработки формы в СУБД	Опрос	«Зачтено» «Не зачтено»
6	Тема 6. Разработка программ сложной ветвящейся структуры и использование массивов данных	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - основные элементы программного кода Уметь: - исследовать возможности разработки программ сложной структуры с управлением массивами в электронной таблице Владеть: - навыками исследования возможности разработки программ сложной структуры с управлением массивами в СУБД	Доклад	«Зачтено» «Не зачтено»
7	Тема 7. Разработка специализированных модулей и процедур	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - понятие специализированного модуля или процедуры Уметь: - применять методы разработки специализированных модулей и процедур в программах офисного пакта Владеть: - навыками исследования возможностей встроенного языка по разработке бизнес приложений и игровых программ	Опрос	«Зачтено» «Не зачтено»
8	Тема 8. Автоматически исполняемые макросы встроенного языка офисных программ	ПК-2 (ИПК-2.1 ИПК-2.2) ПК-4 (ИПК-4.1 ИПК-4.2 ИПК-4.3)	Знать: - понятие макроса Уметь: - исследовать возможности использования автоматических программных процедур для защиты документов офисных программ от несанкционированного доступа Владеть: -алгоритмом решения	Опрос	«Зачтено» «Не зачтено»

			задач		
ИТОГО			Форма контроля	Оценочные средства промежуточной аттестации	Шкала оценивания
			Зачёт с оценкой	Письменный ответ на билет	«Отлично», «Хорошо», «Удовлетворительно», «Неудовлетворительно»

Критерии оценивания результатов обучения для текущего контроля успеваемости и промежуточной аттестации по дисциплине

1. Критерий оценивания опроса:

- зачтено – выставляется обучающемуся, если демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки; освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе; достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности; показывает всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их;

- не зачтено – выставляется обучающемуся, если демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем; слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки; допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки; выставляется обучающемуся, ответ которого содержит существенные пробелы в знаниях основного содержания рабочей программы дисциплины.

2. Критерий доклада:

- зачтено – представленный доклад соответствует тематике, экономически обоснован, выводы по изученной проблеме изложены

логически, соблюдены требования, при разработке доклада были использованы современны информационные технологии;

- не зачтено – доклад обучающимся не представлена; материалы доклад не обоснованы или логически не связаны, использованы устаревшие источники информации.

3. Критерии оценивания письменного ответа на билет на зечете с оценкой:

- отлично – выставляется обучающемуся, если: даны исчерпывающие и обоснованные ответы на все поставленные вопросы, правильно и рационально (с использованием рациональных методик) решены соответствующие задачи; в ответах выделялось главное, все теоретические положения умело увязывались с требованиями руководящих документов; ответы были четкими и краткими, а мысли излагались в логической последовательности; показано умение самостоятельно анализировать факты, события, явления, процессы в их взаимосвязи и диалектическом развитии;

- хорошо – выставляется обучающемуся, если: даны полные, достаточно обоснованные ответы на поставленные вопросы, правильно решены практические задания; в ответах не всегда выделялось главное, отдельные положения недостаточно увязывались с требованиями руководящих документов, при решении практических задач не всегда использовались рациональные методики расчётов; ответы в основном были краткими, но не всегда четкими; показано слабое умение самостоятельно анализировать факты, события, явления, процессы в их взаимосвязи и диалектическом развитии;

- удовлетворительно – выставляется обучающемуся, если: даны в основном правильные ответы на все поставленные вопросы, но без должной глубины и обоснования, при решении практических задач обучающийся использовал прежний опыт и не применял новые методики выполнения расчётов, однако на уточняющие вопросы даны в целом правильные ответы; при ответах не выделялось главное; отдельные положения недостаточно увязывались с требованиями руководящих документов, при решении практических задач не использовались рациональные методики расчётов; ответы были многословными, нечеткими и без должной логической последовательности, на отдельные дополнительные вопросы не даны положительные ответы; показано неумение самостоятельно анализировать факты, события, явления, процессы в их взаимосвязи и диалектическом развитии.

- неудовлетворительно – выставляется обучающемуся, если не выполнены требования, соответствующие оценке “удовлетворительно”.

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

3.1. Вопросы для проведения опроса:

1. Сущность и содержание объектно-ориентированной технологии программирования, общие сведения и особенности языка программирования офисных приложений Visual Basic for Application (VBA).
2. Понятие объекта программирования в VBA и его свойств, принципы взаимодействия программных объектов, назначение основных элементов интерфейса и панелей инструментов VBA.
3. Понятие программного модуля и основные типы модулей встроенного языка, процедура и её структура. Событие и его свойства.
4. Понятие макроса и его интерпретация в VBA, макрорекордер и его состав, а также основные методы, и приёмы работы с ним.
5. Разработка макроса с помощью макрорекордера, исследование возможностей по изменению структуры и управления макросом.
6. Понятие программного кода, исследование возможностей по редактированию макроса, основные способы запуска макроса на исполнение, отладка макроса.
7. Основные отличия в разработке программного кода специализированных процедур и функций в программах офисного пакета.
8. Понятие модуля и основные типы модулей в VBA.
9. Понятие процедуры и её структура в VBA, правила записи операторов в процедуре, инструкция, переменные и типы переменных в VBA, литералы, константы и идентификаторы, выражения и операции в VBA.
10. Определение функций и процедур в VBA.
11. Возможности по вводу и выводу данных в текстовом редакторе, электронной таблице и СУБД.
12. Алгоритм программных процедур реализующих простейшие арифметические операции.
13. Возможности кодирования и отладки программных процедур реализующих простейшие арифметические операции в электронной таблице.
14. Исследование возможности модификации программы для выполнения арифметических операций в текстовом редакторе.
15. Возможности модификации программы для выполнения арифметических операций в СУБД.
16. Понятие графического интерфейса программы, форма и её основные свойства, способы и последовательность разработки графического интерфейса в форме VBA.
17. Основные инструменты, применяемые при разработке формы в VBA.

18. Возможности разработки формы в электронной таблице.
19. Возможности разработки формы в текстовом редакторе.
20. Возможности разработки формы в СУБД.
21. Основные элементы программного кода, обеспечивающие формирование ветвящейся структуры программы.
22. Массивы, виды массивов и управление массивами информации.
23. Возможности разработки программ сложной структуры с управлением массивами в электронной таблице.
24. Возможности разработки программ сложной структуры с управлением массивами в текстовом редакторе.
25. Возможности разработки программ сложной структуры с управлением массивами в СУБД.
26. Понятие специализированного модуля или процедуры.
27. Методы разработки специализированных модулей и процедур в программах офисного пакта.
28. Возможностей встроенного языка по разработке бизнес приложений и игровых программ.
29. Понятие макроса или программной процедуры исполняемой автоматически.
30. Возможности использования автоматических программных процедур для защиты документов офисных программ от несанкционированного доступа.
31. Понятие вируса документов пакета офисных программ и исследование возможности разработки вирусной программы.

3.2. Примерный перечень тем докладов и сообщений:

1. Сущность и содержание объектно-ориентированной технологии программирования, общие сведения и особенности языка программирования офисных приложений Visual Basic for Application (VBA).
2. Понятие объекта программирования в VBA и его свойств, принципы взаимодействия программных объектов, назначение основных элементов интерфейса и панелей инструментов VBA.
3. Понятие программного модуля и основные типы модулей встроенного языка, процедура и её структура. Событие и его свойства.
4. Понятие макроса и его интерпретация в VBA, макрорекордер и его состав, а также основные методы, и приёмы работы с ним.
5. Разработка макроса с помощью макрорекордера, исследование возможностей по изменению структуры и управления макросом.
6. Понятие программного кода, исследование возможностей по редактированию макроса, основные способы запуска макроса на исполнение, отладка макроса.
7. Основные отличия в разработке программного кода специализированных процедур и функций в программах офисного пакета.

8. Понятие модуля и основные типы модулей в VBA.

9. Понятие процедуры и её структура в VBA, правила записи операторов в процедуре, инструкция, переменные и типы переменных в VBA, литералы, константы и идентификаторы, выражения и операции в VBA.

10. Определение функций и процедур в VBA.

11. Исследование возможностей по вводу и выводу данных в текстовом редакторе, электронной таблице и СУБД.

12. Разработка алгоритма программных процедур реализующих простейшие арифметические операции.

13. Исследование возможности кодирования и отладки программных процедур реализующих простейшие арифметические операции в электронной таблице.

14. Исследование возможности модификации программы для выполнения арифметических операций в текстовом редакторе.

15. Исследование возможности модификации программы для выполнения арифметических операций в СУБД.

16. Понятие графического интерфейса программы, форма и её основные свойства, способы и последовательность разработки графического интерфейса в форме VBA.

17. Основные инструменты, применяемые при разработке формы в VBA.

18. Исследование возможности разработки формы в электронной таблице.

19. Исследование возможности разработки формы в текстовом редакторе.

20. Исследование возможности разработки формы в СУБД.

21. Основные элементы программного кода, обеспечивающие формирование ветвящейся структуры программы.

22. Массивы, виды массивов и управление массивами информации.

23. Исследование возможности разработки программ сложной структуры с управлением массивами в электронной таблице.

24. Исследование возможности разработки программ сложной структуры с управлением массивами в текстовом редакторе.

25. Исследование возможности разработки программ сложной структуры с управлением массивами в СУБД.

3.3. Вопросы для проведения зачета с оценкой:

1. Понятие объекта программирования в VBA и его свойств, принципы взаимодействия программных объектов, назначение основных элементов интерфейса и панелей инструментов VBA.

2. Понятие программного кода, исследование возможностей по редактированию макроса, основные способы запуска макроса на исполнение, отладка макроса.

3. Алгоритм программных процедур реализующих простейшие арифметические операции.

4. Возможности модификации программы для выполнения арифметических операций в СУБД.

5. Основные элементы программного кода, обеспечивающие формирование ветвящейся структуры программы.

6. Методы разработки специализированных модулей и процедур в программах офисного пакта.

7. Понятие специализированного модуля или процедуры.

8. Понятие вируса документов пакета офисных программ и исследование возможности разработки вирусной программы.

9. Возможности использования автоматических программных процедур для защиты документов офисных программ от несанкционированного доступа.

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Зачет с оценкой является заключительным этапом процесса формирования компетенций обучающегося при изучении дисциплины и имеет целью проверку и оценку знаний обучающегося по теории и применению полученных знаний, умений и навыков при решении практических задач.

Зачет с оценкой проводится по расписанию, сформированному учебно-методическим управлением, в сроки, предусмотренные календарным учебным графиком.

Зачет с оценкой принимается преподавателем, ведущим лекционные занятия.

Зачет с оценкой проводится только при предъявлении обучающимся зачетной книжки и при условии выполнения всех контрольных мероприятий, предусмотренных учебным планом и рабочей программой дисциплины.

Обучающимся на зачете с оценкой представляется право выбрать один из билетов. Время подготовки к ответу составляет 30 минут. По истечении установленного времени обучающийся должен ответить на вопросы билета.

Результаты зачета с оценкой оцениваются по четырехбалльной системе и заносятся в зачетно-экзаменационную ведомость и зачетную книжку. В зачетную книжку заносятся только положительные оценки. Подписанный преподавателем экземпляр ведомости сдаётся не позднее следующего дня в деканат.

В случае неявки обучающегося на зачет с оценкой, экзамен в зачетно-экзаменационную ведомость делается отметка «неявка».

Обучающиеся, не прошедшие промежуточную аттестацию по дисциплине, должны ликвидировать академическую задолженность в установленном локальными нормативными актами Института порядке.

5. Материалы для компьютерного тестирования обучающихся в рамках проведения контроля наличия у обучающихся сформированных результатов обучения по дисциплине

Общие критерии оценивания

№ п/п	Процент правильных ответов	Оценка
1	86 % – 100 %	5 («отлично»)
2	70 % – 85 %	4 («хорошо»)
3	51 % – 69 %	3 («удовлетворительно»)
4	50 % и менее	2 («неудовлетворительно»)

Вариант 1

Номер вопроса и проверка сформированной компетенции

№ вопроса	Код компетенции	№ вопроса	Код компетенции
1	ПК-2 (ИПК-2.1, ИПК-2.2)	11	ПК-2 (ИПК-2.1, ИПК-2.2)
2	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	12	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
3	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	13	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
4	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	14	ПК-2 (ИПК-2.1, ИПК-2.2)
5	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	15	ПК-2 (ИПК-2.1, ИПК-2.2)
6	ПК-2 (ИПК-2.1, ИПК-2.2)	16	ПК-2 (ИПК-2.1, ИПК-2.2)
7	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	17	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
8	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	18	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
9	ПК-2 (ИПК-2.1, ИПК-2.2)	19	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
10	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	20	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)

Ключ ответов

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

Задание № 1.

Когда необходимо составлять блок-схему программы?

Ответ:

1. До начала составления самой программы

2. После составления программы

3. В процессе составления программы

Задание № 2.

Наиболее наглядной формой описания алгоритма является структурно-стилизованный метод:

Ответ:

1. представление алгоритма в виде схемы

2. язык программирования высокого уровня

3. словесное описание алгоритма

Задание № 3.

Перевод программ с языка высокого уровня на язык более низкого уровня обеспечивает программа:

Ответ:

1. ассемблер

2. паскаль

3. Компилятор

4. Фортран

Задание № 4.

В графических схемах алгоритмов стрелки направлений на линиях потоков:

Ответ:

1. необходимо рисовать, если направление потока сверху вниз и слева направо

2. рисовать не нужно

3. Необходимо рисовать, если направление потока снизу вверх и справа налево

Задание № 5.

Разработка алгоритма решения задачи – это:

Ответ:

1. сведение задачи к математической модели, для которой известен метод решения
2. выбор наилучшего метода из имеющихся
3. точное описание данных, условий задачи и ее целого решения
4. определение последовательности действий, ведущих к получению результатов

Задание № 6.

Языком высокого уровня является:

Ответ:

1. Макроассемблер
2. Шестнадцатеричный язык
3. Фортран
4. Ассемблер

Задание № 7.

Алгоритм, в котором действия выполняются друг за другом, не повторяясь называется:

Ответ:

1. Линейным
2. циклическим
3. разветвленным
4. Простым

Задание № 8.

Разработке алгоритма предшествует:

1. постановка задачи, разработка математической модели, проектирование программ
2. постановка задачи, разработка математической модели
3. постановка задачи, выбор метода решения, проектирование программ
4. постановка задачи, разработка математической модели, выбор метода решения

Задание № 9.

Символьный тип данных объявляется служебным словом:

Ответ:

1. CHAR;
2. STRING;
3. BYTE;
4. WORD;

Задание № 10.

В операторе присваивания summa := sqr(x)+3*a переменными являются:

Ответ:

1. a, x, summa

2. x, a

3. sqr,x,a

4. summa, sqr, x, a

Задание № 11.

Процедура INC(x,k):

Ответ:

1. преобразует десятичное число x в строку из k символов

2. увеличивает значение переменной x на величину k

3. уменьшает значение переменной x на величину k

4. преобразует строку символов x в число, содержащее k десятичных знаков

Задание № 12.

Если функция имеет тип void, то неверно, что:

Ответ:

1. она не может содержать оператор return;

2. оператор return в теле функции не обязателен;

3. выражение в операторе return опускается;

4. функция не возвращает никакого значения.

Задание № 13.

Когда компилятор встраивает код функции вместо ее вызова, такая функция называется:

Ответ:

1. конструктором

2. деструктором

3. Подставляемой

4. рекурсивной

Задание № 14.

Цель перегрузки функции состоит в:

Ответ:

1. том, чтобы функция с одним именем по-разному выполнялась и возвращала разные значения при обращении к ней с разными по типам и по количеству фактическими параметрами

2. создании функции, которые могут обрабатывать разнотипные данные

3. создании другого имени для уже существующей функции

том, чтобы функции с разными именами выполнялись одинаково для одинакового числа параметров

Задание № 15.

Имена параметров, указанные в прототипе, определении и вызове функции:

Ответ:

1. должны соответствовать друг другу, так как все параметры идентифицируются по имени;
2. не должны соответствовать друг другу, так как все параметры идентифицируются позицией;
3. должны соответствовать друг другу, так как все параметры идентифицируются позицией;
4. не должны соответствовать друг другу, так как все параметры идентифицируются по имени;

Задание № 16.

Выберите прототип, объявляющий функцию, аргументы которой передаются как ссылки с помощью ссылок:

Ответ:

1. void change(int &a, &b);
2. void move(int *x, int *y);
3. void print(int &p1, int &p2);
4. void show(int k, int l);

Задание № 17.

Выберите прототип, объявляющий функцию, аргументы которой передаются как значения:

Ответ:

1. void change(int &a, &b);
2. void move(int *x, int *y);
3. void print(int &p1, int &p2);
4. void show(int k, int l);

Задание № 18.

Функция … выполняет конкатенацию двух строк:

Ответ:

1. strstr(s, t)
2. strcmp(s, t)
3. strcat(s, t)
4. strcpy(s, t)
5. stradd(s, t)

Задание № 19.

Функция … выполняет сравнение двух строк:

Ответ:

1. strstr(s, t)

2. strcmp(s, t)
3. strcat(s, t)
4. strcpy(s, t)
5. stradd(s, t)

Задание № 20.

Функция … возвращает длину строки:

Ответ:
1. str(s)
2. GetLen(s)
3. length(s)
4. strlength(s)
5. strlen(s)

Вариант 2

Номер вопроса и проверка сформированной компетенции

№ вопроса	Код компетенции	№ вопроса	Код компетенции
1	ПК-2 (ИПК-2.1, ИПК-2.2)	11	ПК-2 (ИПК-2.1, ИПК-2.2)
2	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	12	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
3	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	13	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
4	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	14	ПК-2 (ИПК-2.1, ИПК-2.2)
5	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	15	ПК-2 (ИПК-2.1, ИПК-2.2)
6	ПК-2 (ИПК-2.1, ИПК-2.2)	16	ПК-2 (ИПК-2.1, ИПК-2.2)
7	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	17	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
8	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	18	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
9	ПК-2 (ИПК-2.1, ИПК-2.2)	19	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
10	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	20	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)

Ключ ответов

№ вопроса	Верный ответ	№ вопроса	Верный ответ
1	2	11	2,4
2	2	12	4
3	4	13	3
4	1	14	1
5	1	15	2
6	2	16	3
7	2	17	2
8	2	18	3
9	1	19	3
10	4	20	2,4

Задание № 1.

Объектно-ориентированное программирование характеризуется:

Ответ:

1. Наличием одной линейной программы
2. Разделением программы на модули
3. все данные об объекте, его связи с другими объектами объединяются в одну структурную переменную

Задание № 2.

Метод это:

Ответ:

1. Функция, получая в качестве обязательного параметра указатель на объект
2. Структура, хранящая указатели this, parrent, supper
3. Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии
4. Определенный программистом абстрактный тип данных

Задание № 3.

Объект это:

Ответ:

1. Функция, получая в качестве обязательного параметра указатель на объект
2. Структура, хранящая указатели this, parrent, supper
3. Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии
4. Определенный программистом абстрактный тип данных

Задание № 4.

Класс это:

Ответ:

1. Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии
2. Определенный программистом абстрактный тип данных

3. Переменная, описанная абстрактным типом данных

Задание № 5.

Для выполнения операции присвоение одного объекта другому:

Ответ:

1. Достаточно чтобы в классе этих объектов был специальный конструктов — конструктор копирования

2. Классы этих объектов нужно описать одинаковым образом

3. Достаточно чтобы в классе этих объектов были конструкторы и деконструкторы

4. Необходимо описать оба объекта одним именем класса и иметь в классе конструктор копирования

Задание № 6.

В каких случаях надо иметь в классе конструктор копирования:

Ответ:

1. Когда нужно передать элементы класса

2. Для выполнения операции присвоение одного элемента другому

3. Для создания дружественной функции

4. Для уничтожения объекта из памяти

Задание № 7.

В чем состоит задача конструктора копирования:

Ответ:

1. Четко определить порядок создания копии объекта

2. Размножать объект в указанном количестве копий

3. Копировать свойства других объектов

4. Создать this указатель

Задание № 8.

Какое ключевое слово должно быть использовано, чтобы указать что объект не является модифицируемым и любая попытка изменения этого типа является ошибкой:

Ответ:

1. readonly

2. const

3. static

4. undeclare

Задание № 9.

Как называются элементы класса, которые относятся ко всем экземплярам объектов класса:

Ответ:

1. Статические

2. Динамические

3. Константные

4. Защищенные

Задание № 10.

В каких случаях следует вводить статические элементы массива:

Ответ:

1. Когда требуется контроль количества общих методов класса
2. Когда требуется доступ к элементам массива из объектов дружественного класса
3. Когда требуется одновременный доступ ко всем объектам класса
4. Когда требуется определить изменяемые элементы класса

Задание № 11.

Структура объединяет:

Ответ:

1. данные одного типа
2. логически связанные данные
3. целые именованные значения
4. переменные

Задание № 12.

При определении структуры:

Ответ:

1. выделяется память под переменную
2. выделяется память под поля
3. выделяется память под структуру
4. выделяется память после определения структурной переменной

Задание № 13.

При обращении к полю структуры левым операндом операции (.) является:

Ответ:

1. поле структуры;
2. имя структуры;
3. структурная переменная;
4. ключевое слово struct.

Задание № 14.

Истинно ли следующее утверждение: вы можете присвоить значение одной структурной переменной другой структурной переменной того же типа:

Ответ:

1. истинно
2. Нет

Задание № 15.

Для объявления группы данных различных типов используется ключевое слово:

Ответ:
1. enum
2. struct
3. typedef
4. union
5. coalition

Задание № 16.

Структура отличается от класса тем, что все ее члены:

Ответ:
1. не являются функциями
2. могут быть только базового типа
3. являются общими
4. являются защищенными

Задание № 17.

Объединенное в единое целое множество поименованных элементов разных типов есть

Ответ:
1. массив
2. структура
3. класс
4. файл
5. ссылка

Задание № 18.

Следующий оператор: struct { char N[12]; int value;} XX, YY, EE[8], *pst; содержит определение….структурного типа.

Ответ:
1. 3х структур XX, YY, EE и указателя на тип char.
2. 3х структур XX, YY, EE и указателя на тип int.
3. 2х структур XX и YY, массива структур EE и указателя pst на структуру.

Задание № 19.

Уточненное имя используется для

Ответ:
1. обращения к локальным переменным,
2. обращения к глобальным переменным,
3. обращения к элементам структуры,
4. обращения к функции,
5. нет правильного ответа.

Задание № 20.

Наиболее важным из назначений функции является:

Ответ:

1. именование блока операторов;
2. уменьшение размера программы;
3. обработка аргументов и возвращение значения;
4. структуризация программы.

Вариант 3

Номер вопроса и проверка сформированной компетенции

№ вопроса	Код компетенции	№ вопроса	Код компетенции
1	ПК-2 (ИПК-2.1, ИПК-2.2)	11	ПК-2 (ИПК-2.1, ИПК-2.2)
2	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	12	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
3	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	13	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
4	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	14	ПК-2 (ИПК-2.1, ИПК-2.2)
5	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	15	ПК-2 (ИПК-2.1, ИПК-2.2)
6	ПК-2 (ИПК-2.1, ИПК-2.2)	16	ПК-2 (ИПК-2.1, ИПК-2.2)
7	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	17	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
8	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	18	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
9	ПК-2 (ИПК-2.1, ИПК-2.2)	19	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
10	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	20	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)

Ключ ответов

№ вопроса	Верный ответ	№ вопроса	Верный ответ
1	3	11	4
2	2	12	3
3	1,2,3	13	4
4	1	14	4
5	1	15	2
6	4	16	4

7	4	17	2
8	4	18	2
9	3	19	4
10	1	20	1

Задание № 1.

Аргумент функции – это:

Ответ:

1. переменная функции, получающая значение из вызывающей программы;
2. способ, с помощью которого функция защищает себя от воздействия значений, передаваемых вызывающей программой;
3. значение, передаваемое вызывающей программой в функцию;
4. значение, возвращаемое функцией вызывающей программе.

Задание № 2.

Истинно ли следующее утверждение: когда аргументы передаются по значению, функция имеет доступ к переменным вызывающей программы?

Ответ:

1. истинно
2. ложно
3. зависит от сигнатуры функции

Задание № 3.

Какие из перечисленных ниже элементов программы можно передавать в функцию:

Ответ:

1. константы;
2. переменные;
3. структуры;
4. заголовочные файлы.

Задание № 4.

Истинно ли следующее утверждение: когда функция возвращает значение, ее вызов можно ставить справа от знака операции присваивания?

Ответ:

1. истинно
2. ложно
3. ошибка компиляции
4. если функция не имеет аргументов

Задание № 5.

Где указывается тип значения, возвращаемый функцией?

Ответ:

1. в начале объявления и описателя

2. в теле функции
3. в операторе вызова
4. в операторе return

Задание № 6.

Какова причина использования ссылочного механизма передачи аргументов в функцию?

Ответ:
1. убыстряется выполнение функции
2. защита от изменений функцией значений аргументов
3. легко передаются аргументы в функцию
4. изменение исходного значения аргумента
5. возможность программе обращаться к копии переменной

Задание № 7.

Цикл с предусловием определяется служебным словом

Ответ:

1. FOR;
2. WHILE;
3. REPEAT;
4. IF;

Задание № 8.

Цикл с постусловием определяется служебным словом

Ответ:

1. FOR;
2. WHILE;
3. CASE;
4. REPEAT.

Задание № 9.

Неверно записано описание переменных

Ответ:

1. VAR A,B:real; I,J:integer;
2. VAR j,g,t:integer; i:real;
3. VAR i,max:integer; a,max:real;
4. VAR a,b:real; c,d:real;

Задание № 10.

В Паскале различие в высоте букв (прописные или строчные)

Ответ:

1. имеет значение при написании текстовых констант+
2. имеет значение для написания служебных слов
3. имеет значение для написания служебных слов
4. игнорируется

Задание № 11.

Какое выражение верно?

Ответ:

1. Структура подпрограммы аналогична структуре основной программы
2. Подпрограмма предназначена для выполнения какой-то последовательности действий
3. В Паскале имеется две разновидности подпрограмм – процедуры и функции
4. В подпрограмме нельзя использовать глобальные переменные+

Задание № 12.

Подпрограммы располагаются в разделе описания?

Ответ:

1. переменных
2. типов
3. процедур и функций
4. меток

Задание № 13.

Каких параметров подпрограммы не существует?

Ответ:

1. параметров-переменных
2. параметров-констант
3. параметров-значений
4. параметров-модулей

Задание № 14.

Как называются имена представленные в разделе описания основной программы?

Ответ:

1. формальными
2. параметрами-константами
3. локальными
4. Глобальными

Задание № 15.

Какие виды подпрограмм есть в паскале

Ответ:

1. модуль и функция
2. процедура и функция
3. модуль и процедура
4. процедура

Задание № 16.

Где правильно описана процедура?

Ответ:

1. PROCEDURA (a:array [1..10] of integer: b:real)
2. PROSEDURA (var a:array [1..10] of integer; b:real)
3. PROSEDURA (a:array of integer; b:real)
4. PROCEDURE (a:array of integer; var b:real)

Задание № 17.

Какое служебное слово указывается в заголовке процедуры?

Ответ:

1. BEGIN
2. PROCEDURE
3. FUNCTION
4. PROCEDURA

Задание № 18.

Какие объекты можно использовать в качестве формальных параметров в процедуре?

Ответ:

1. константы, переменные, выражения
2. переменные с указанием их типа
3. выражения, константы
4. типизированные константы

Задание № 19.

Что представляют собой фактические параметры?

Ответ:
1. константы, переменные, выражения с указанием их типа константы, переменные, выражения с указанием их типа
2. переменные с указанием их типа
3. выражения, константы

Задание № 20.

Каким словом разделяется перечень однотипных формальных переменных в процедуре?

Ответ:

1. Запятой
2. точкой с запятой
3. двоеточием
4. точкой

Вариант 4

Номер вопроса и проверка сформированной компетенции

№ вопроса	Код компетенции	№ вопроса	Код компетенции
1	ПК-2 (ИПК-2.1, ИПК-2.2)	11	ПК-2 (ИПК-2.1, ИПК-2.2)
2	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	12	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
3	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	13	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
4	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	14	ПК-2 (ИПК-2.1, ИПК-2.2)

5	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	15	ПК-2 (ИПК-2.1, ИПК-2.2)
6	ПК-2 (ИПК-2.1, ИПК-2.2)	16	ПК-2 (ИПК-2.1, ИПК-2.2)
7	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	17	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
8	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	18	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
9	ПК-2 (ИПК-2.1, ИПК-2.2)	19	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)
10	ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)	20	ПК-2 (ИПК-2.1, ИПК-2.2), ПК-4 (ИПК-4.1, ИПК-4.2, ИПК-4.3)

Ключ ответов

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

Задание № 1.

Аргумент функции – это:

Ответ:
1. переменная функции, получающая значение из вызывающей программы;
2. способ, с помощью которого функция защищает себя от воздействия значений, передаваемых вызывающей программой;
3. значение, передаваемое вызывающей программой в функцию;
4. значение, возвращаемое функцией вызывающей программе.

Задание № 2.

Истинно ли следующее утверждение: когда аргументы передаются по значению, функция имеет доступ к переменным вызывающей программы?

Ответ:
1. истинно
2. ложно
3. зависит от сигнатуры функции

Задание № 3.

Какие из перечисленных ниже элементов программы можно передавать в функцию:

Ответ:
1. константы;
2. переменные;
3. структуры;
4. заголовочные файлы.

Задание № 4.

Истинно ли следующее утверждение: когда функция возвращает значение, ее вызов можно ставить справа от знака операции присваивания?

Ответ:
1. истинно
2. ложно
3. ошибка компиляции
4. если функция не имеет аргументов

Задание № 5.

Где указывается тип значения, возвращаемый функцией?

Ответ:
1. в начале объявления и описателя
2. в теле функции
3. в операторе вызова
4. в операторе return

Задание № 6.

Когда аргумент передается в функцию по ссылке

Ответ:
1. внутри функции создается переменная, хранящая значение этого аргумента;
2. функция не имеет доступа к значению аргумента;
3. в вызывающей программе создается временная переменная для хранения значения аргумента;
4. функция получает доступ к аргументу в вызывающей программе.

Задание № 7.

В каких случаях следует вводить статические элементы массива:

Ответ:
1. Когда требуется контроль количества общих методов класса
2. Когда требуется доступ к элементам массива из объектов дружественного класса
3. Когда требуется одновременный доступ ко всем объектам класса
4. Когда требуется определить изменяемые элементы класса

Задание № 8.

Структура объединяет:

Ответ:
1. данные одного типа
2. логически связанные данные
3. целые именованные значения
4. переменные

Задание № 9.

При определении структуры:

Ответ:
1. выделяется память под переменную
2. выделяется память под поля
3. выделяется память под структуру
4. выделяется память после определения структурной переменной

Задание № 10.

При обращении к полю структуры левым операндом операции (.) является:

Ответ:
1. поле структуры;
2. имя структуры;
3. структурная переменная;
4. ключевое слово struct.

Задание № 11.

Истинно ли следующее утверждение: вы можете присвоить значение одной структурной переменной другой структурной переменной того же типа:

Ответ:

1. истинно

2. Нет

Задание № 12.

Для объявления группы данных различных типов используется ключевое слово:

Ответ:

1. enum
2. struct
3. typedef
4. union
5. coalition

Задание № 13.

Структура отличается от класса тем, что все ее члены:

Ответ:

1. не являются функциями
2. могут быть только базового типа
3. являются общими
4. являются защищенными

Задание № 14.

Объединенное в единое целое множество поименованных элементов разных типов есть

Ответ:

1. массив
2. структура
3. класс
4. файл
5. ссылка

Задание № 15.

Следующий оператор: struct { char N[12]; int value;} XX, YY, EE[8], *pst; содержит определение….структурного типа.

Ответ:

1. 3х структур XX, YY, EE и указателя на тип char.
2. 3х структур XX, YY, EE и указателя на тип int.
3. 2х структур XX и YY, массива структур EE и указателя pst на структуру.

Задание № 16.

Уточненное имя используется для

Ответ:

1. обращения к локальным переменным,
2. обращения к глобальным переменным,
3. обращения к элементам структуры,
4. обращения к функции,
5. нет правильного ответа.

Задание № 17.

Наиболее важным из назначений функции является:

Ответ:

1. именование блока операторов;
2. уменьшение размера программы;
3. обработка аргументов и возвращение значения;
4. структуризация программы.

Задание № 18.

Когда компилятор встраивает код функции вместо ее вызова, такая функция называется:

Ответ:

1. конструктором
2. деструктором
3. Подставляемой
4. рекурсивной

Задание № 19.

Цель перегрузки функции состоит в:

Ответ:

1. том, чтобы функция с одним именем по-разному выполнялась и возвращала разные значения при обращении к ней с разными по типам и по количеству фактическими параметрами
2. создании функции, которые могут обрабатывать разнотипные данные
3. создании другого имени для уже существующей функции
том, чтобы функции с разными именами выполнялись одинаково для одинакового числа параметров

Задание № 20.

Имена параметров, указанные в прототипе, определении и вызове функции:

Ответ:

1. должны соответствовать друг другу, так как все параметры идентифицируются по имени;
2. не должны соответствовать друг другу, так как все параметры идентифицируются позицией;
3. должны соответствовать друг другу, так как все параметры идентифицируются позицией;

4. не должны соответствовать друг другу, так как все параметры идентифицируются по имени;

6. Практические задачи.

Задача 1.

Исследование возможностей по управлению макросом и изменению его структуры

План:

1. Открыть текстовый редактор из состава пакета офисных программ, перейти в меню «Сервис», «Макрос», «Начать запись», запустить макрорекордер языка VBA и создать макрос.
2. Назначить кнопки управления макросом, перейти в меню «Сервис», «Макрос», «Прекратить запись» и сохранить макрос.
3. С помощью редактора языка VBA исследовать состав программного кода макроса, изменить его и сохранить.
4. Проверить его функционирование.

Задача 2.

Понятие модуля и процедуры, а также их структура в VBA.

План:

1. Открыть текстовый редактор из состава пакета офисных программ, перейти в меню «Сервис», «Макросы» и открыть редактор языка VBA.
2. В программном модуле языка VBA самостоятельно разработать процедуру.
3. Проверить его функционирование.

Задача 3.

Исследование возможности разработки процедуры в электронной таблице.

План:

1. Открыть Excel из состава пакета офисных программ, перейти в меню «Сервис», «Макросы» и открыть редактор языка VBA.
2. В программном модуле языка VBA разработать процедуру по расчету и заполнению ячеек в текущей книге Excel.
3. Проверить ее функционирование.

Задача 4.

Исследование возможности разработки процедуры в текстовом редакторе и в СУБД.

План:

1. Открыть СУБД MS ACCESS из состава пакета офисных программ, перейти в меню «Сервис», «Макросы» и открыть редактор языка VBA.
2. В программном модуле языка VBA разработать процедуру по выполнению запроса к таблице.
3. Проверить ее функционирование.

Задача 5.

Назначение элементов интерфейса и структуры встроенного языка программирования пакета офисных программ.

План:

1. Открыть текстовый редактор из состава пакета офисных программ, перейти в меню «Сервис», «Макросы» и открыть редактор языка VBA.
2. Изучить состав интерфейса VBA, создать макрос и сохранить.
3. Проверить его функционирование.

Задача 6.

Исследование возможности настройки интерфейса и изменения структуры встроенного языка VBA.

План:

1. Открыть электронную таблицу из состава пакета офисных программ, перейти в меню «Сервис», «Макросы» и открыть редактор языка VBA.
2. Провести настройку элементов интерфейса языка VBA и исследовать его возможности по разработке специализированных процедур.
3. Создать проект в VBA, сохранить проект.

Задача 7.

Макрорекордер, его состав, основные методы и приёмы работы с ним.

План:

1. Открыть СУБД из состава пакета офисных программ, перейти в меню «Сервис», «Макрос», «Начать запись», запустить макрорекордер языка VBA и создать макрос.
2. Изучить состав макрорекордера, перейти в меню «Сервис», «Макрос», «Прекратить запись» и сохранить макрос.
3. Разработать с помощью макрорекордера макрос и сохранить его.
4. Проверить его функционирование.

Задача 8.

Назначение элементов интерфейса и структуры встроенного языка программирования пакета офисных программ.

План:

1. Открыть текстовый редактор из состава пакета офисных программ, перейти в меню «Сервис», «Макросы» и открыть редактор языка VBA.
2. Изучить состав интерфейса VBA, создать макрос и сохранить. Проверить его функционирование.