

АВТНОМНАЯ НЕКОММЕРЧЕСКАЯ ОРГАНИЗАЦИЯ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ СОЦИАЛЬНЫЙ ИНСТИТУТ»

Утверждаю
Декан факультета



Ж.В. Игнатенко

« 21 » сент 10 20 20 г.

**Комплект оценочных материалов
по дисциплине**

Теория алгоритмов

основной образовательной программы
по специальности СПО

09.02.03 Программирование в компьютерных системах

Базовый уровень подготовки

год начала подготовки – 2020

Рекомендовано
на заседании кафедры
от « 21 » 10 2020 г.
протокол № 2
Зав. кафедрой Ж.В. Игнатенко

Согласовано
Заведующий выпускающей
кафедрой
Ж.В. Игнатенко

Одобрено
на заседании учебно-методической
комиссии факультета
от « 21 » 10 2020 г.
протокол № 2
Председатель УМК Ж.В. Игнатенко

Ставрополь, 2020 г.

Комплект оценочных материалов разработан на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности СПО 09.02.03 Программирование в компьютерных системах базового уровня и рабочей программы дисциплины «Теория алгоритмов»

Разработчик(и):

АНО ВО СКСИ

преподаватель кафедры
прикладной информатики и
математики

О.В. Краскова

СОДЕРЖАНИЕ

1. Паспорт комплекта контрольно-оценочных средств.....	4
2. Результаты освоения дисциплины, подлежащие проверке	4
3. Оценка освоения дисциплины:	6
4. Контрольно-оценочные материалы для промежуточной аттестации по дисциплине	38

1. Паспорт комплекта контрольно-оценочных средств

В результате освоения дисциплины «Теория алгоритмов» обучающийся должен обладать предусмотренными ФГОС СПО по специальности 09.02.03 Программирование в компьютерных системах базового уровня подготовки следующими умениями, знаниями, которые формируют профессиональную компетенцию, и общими компетенциями:

В результате освоения дисциплины обучающийся должен:

ПК 1.1. Выполнять разработку спецификаций отдельных компонент.

ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

Уметь:

У1 разрабатывать алгоритмы для конкретных задач;

У2 определять сложность работы алгоритмов;

Знать:

З1 основные модели алгоритмов;

З2 методы построения алгоритмов

З3 методы вычисления сложности работы алгоритмов

практический опыт: разделом VI ФГОС СПО «Требования к структуре программы подготовки специалистов среднего звена» [таблица 3] не предусмотрен.

Формой аттестации по дисциплине является дифференцированный зачет.

2. Результаты освоения дисциплины, подлежащие проверке

2.1. В результате аттестации по дисциплине осуществляется комплексная проверка следующих умений и знаний, а также динамика формирования общих компетенций:

Таблица 1.1

Результаты обучения: умения, знания и общие компетенции	Показатели оценки результата	Форма контроля и оценивания
Уметь: разрабатывать алгоритмы для		

конкретных задач;		
ПК 1.1. ОК 2. ОК 3.	<p>Умеет выполнять разработку спецификаций отдельных компонент.</p> <p>Умеет организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.</p> <p>Умеет принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.</p>	<p>Практическая работа</p> <p>Самостоятельная работа</p> <p>Тестирование</p>
Знать: основные модели алгоритмов		
ОК 4. ОК5.	<p>Знает как осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.</p> <p>Знает как использовать информационно-коммуникационные технологии в профессиональной деятельности.</p>	<p>Практическая работа</p> <p>Самостоятельная работа</p> <p>Тестирование</p> <p>Письменный опрос</p>
Уметь: определять сложность работы алгоритмов		
ОК 1. ОК 6.	<p>Умеет понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.</p> <p>Умеет работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.</p>	<p>Практическая работа</p> <p>Самостоятельная работа</p> <p>Тестирование</p>
Знать: методы построения алгоритмов;		
ПК 1.2. ОК 7. ОК 8.	<p>Знает как осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.</p> <p>Знает как брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.</p> <p>Знает как самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.</p>	<p>Практическая работа</p> <p>Самостоятельная работа</p> <p>Тестирование.</p> <p>Письменный опрос</p>
Знать: методы вычисления сложности работы алгоритмов;		

ОК 8. ОК 9.	Знает как самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации. Знает как ориентироваться в условиях частой смены технологий в профессиональной деятельности.	Практическая работа Самостоятельная работа Тестирование Письменный опрос
----------------	--	---

3. Оценка освоения дисциплины:

3.1. Формы и методы оценивания

Предметом оценки служат умения и знания, предусмотренные ФГОС по дисциплине «Теория алгоритмов», направленные на формирование общих и профессиональных компетенций.

Проверяемые ОК, умения и знания	Форма контроля
Текущий контроль	
31-33	Практическая работа. Самостоятельная работа. Тестирование. Письменный опрос
У1,У2	Практическая работа. Самостоятельная работа. Тестирование
Промежуточная аттестация	
31-33 У1,У2	Дифференцированный зачет.

3.2. Типовые задания для оценки освоения дисциплины

1) Задания в тестовой форме (пример)

ТЕМА 2. ЛИНЕЙНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ АЛГОРИТМЫ

Тест №1.

Критерии оценки

Предлагаемый пакет оценочных материалов включает два равнозначных комплекта, состоящих из семи заданий.

Все задания репродуктивного характера, предполагают воспроизведение теоретического материала.

Процент правильных ответов	Оценка	Количество правильных ответов
90% – 100%	«Отлично»	9
75% – 89%	«Хорошо»	8 – 6
50% – 74%	«Удовлетворительно»	5 – 4
Менее 50%	«Неудовлетворительно»	Менее 3

1. Алгоритм называется линейным:

- 1) если он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
- 2) если ход его выполнения зависит от истинности тех или иных условий;

- 3) если его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий;
 - 4) если он представим в табличной форме;
 - 5) если он включает в себя вспомогательный алгоритм.
2. В блок-схеме начало и конец алгоритма обозначается фигурой:



Что изменяет операция присваивания?

- 1) значение переменной
- 2) имя переменной
- 3) тип переменной
- 4) тип алгоритма

3. Характерным признаком линейной программы является:

- 1) выполнение операторов в порядке их записи
- 2) наличие в каждой программной строке только одного оператора
- 3) использование в ней исключительно операторов присваивания
- 4) присутствие в ней операторов условного и безусловного перехода
- 5) присутствие в ней операторов цикла

5. Какое свойство алгоритмов характеризует данное определение: «Алгоритм должен приводить к решению за конечное число шагов»:

- 1) Дискретность
- 2) Определенность
- 3) Результативность
- 4) Полнота
- 5) Массовость
- 6) Правильность

6. Какое свойство алгоритмов характеризует данное определение: «Каждое правило алгоритма должно быть четким, однозначным»:

- 1) Дискретность
- 2) Определенность
- 3) Результативность
- 4) Полнота
- 5) Массовость
- 6) Правильность

7. Какое свойство алгоритмов характеризует данное определение: «Алгоритм решения задачи разрабатывается в общем виде, т.е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными»:

- 1) Дискретность
- 2) Определенность
- 3) Результативность
- 4) Полнота
- 5) Массовость
- 6) Правильность

8. Выберите вариант, где представлены все способы описания алгоритма (но нет лишних).

- 1) Словесный, Формульно-словесный, Графический, Программный
- 2) Словесный, Формульный, Формульно-словесный, Графический, Программный
- 3) Словесный, Формульный, Графический, Программный
- 4) Формульный, Графический, Программный

9. Что такое блок схема?

- 1) распространенный тип схем, описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.
- 2) схематичное изображение элементов алгоритма, которые собраны воедино с помощью линий в определенном порядке, демонстрируя алгоритм.
- 3) строгое изображение элементов алгоритма, которые собраны воедино по правилам с
- 4) помощью линий в определенном порядке, демонстрируя алгоритм.
- 5) схема программы на языке программирования, представленная в виде блоков, соединенных линиями

ТЕМА 3. ВЕТВЛЕНИЯ И ЦИКЛЫ В ВЫЧИСЛИТЕЛЬНЫХ АЛГОРИТМАХ

Тест №2

Критерии оценки

Предлагаемый пакет контрольно-измерительных материалов включает два равнозначных комплекта, состоящих из семи заданий.

Все задания репродуктивного характера, предполагают воспроизведение теоретического материала.

Процент правильных ответов	Оценка	Количество правильных ответов
90% – 100%	«Отлично»	10 – 9
75% – 89%	«Хорошо»	8 – 7
50% – 74%	«Удовлетворительно»	6 – 5
Менее 50%	«Неудовлетворительно»	Менее 5

Вариант 1

1. Алгоритмическая структура «ветвление» может содержать

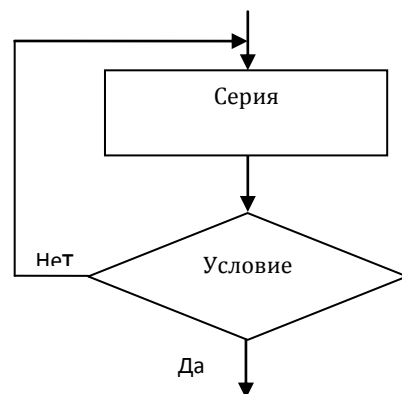
- а) только два варианта действий;
- б) только один вариант действий;
- в) один или два варианта действий;
- г) любое конечное число вариантов действий.

2. Как называется способ многократного выполнения действий в зависимости от заданного условия?

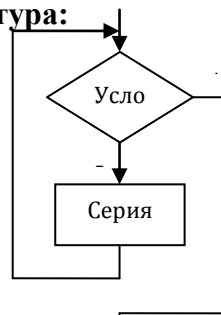
- а) Выбор;
- б) Ветвление;
- в) Цикл;
- г) Линейный алгоритм.

3. На рисунке показана алгоритмическая структура:

1. Следование.
2. Ветвление.
3. Цикл-пока.
4. Цикл-до.
5. Цикл с параметром.

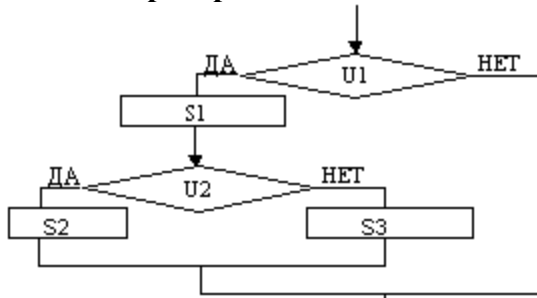


4. На рисунке показана алгоритмическая структура:



1. Следование.
2. Ветвление.
3. Цикл-пока.
4. Цикл-до.
5. Цикл с параметром.

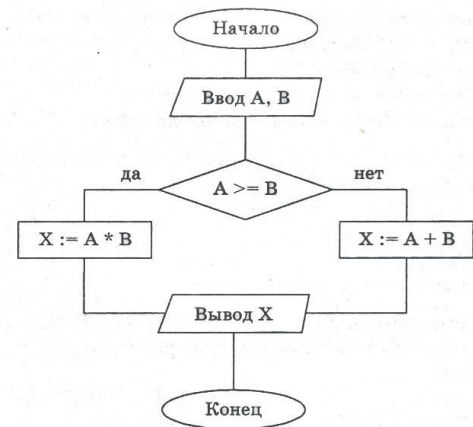
5. U_1, U_2 — обозначают некоторые условия, а S_1, S_2, S_3 — операторы. Выбрать логическое выражение, задающее условие, при котором будет выполняться оператор S_3 .



- а) $U_1 = \text{ИСТИНА}$ ИЛИ $U_2 = \text{ЛОЖЬ}$
- б) $U_1 = \text{ИСТИНА}$
- в) $U_1 = \text{ИСТИНА}$ И $U_2 = \text{ЛОЖЬ}$
- г) $U_1 = \text{ИСТИНА}$ ИЛИ НЕ($U_2 = \text{ИСТИНА}$)

6. При исходных данных $A=5, B=4$ определите результат выполнения алгоритма, изображенного в виде блок-схемы:

Что будет выведено на экран?

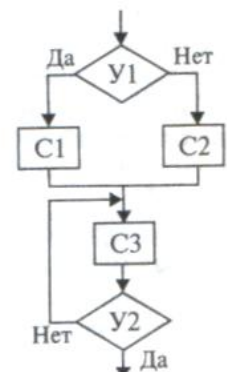


7. Алгоритмическая структура «цикл» предназначена для

- а) многократного выполнения одной и той же последовательности действий;
- б) выполнения одной из двух последовательностей действий в зависимости от значения некоторого логического выражения;
- в) последовательного выполнения действий;
- г) выполнения действий в случае истинности некоторого логического выражения.

8. Определите название блок-схемы:

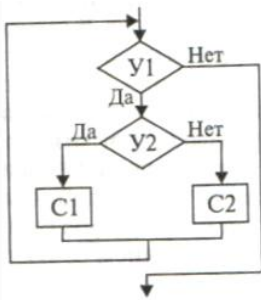
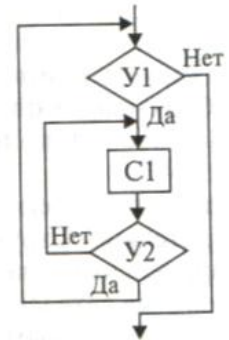
- а) Вложенные ветвления.
- б) Цикл-пока с вложенным ветвлением.
- в) Вложенные циклы-пока.
- г) Следование ветвления и цикла-до.



д) Вложенные циклы. Внешний – цикл-пока, внутренний – цикл-до.

9. Определите название блок-схемы:

- а) Вложенные ветвления.
- б) Цикл-пока с вложенным ветвлением.
- в) Вложенные циклы-пока.
- г) Следование ветвления и цикла-до.
- д) Вложенные циклы. Внешний – цикл-пока, внутренний – цикл-до.



10. Определите название блок-схемы:

Вложенные ветвления. Цикл-пока с вложенным ветвлением.

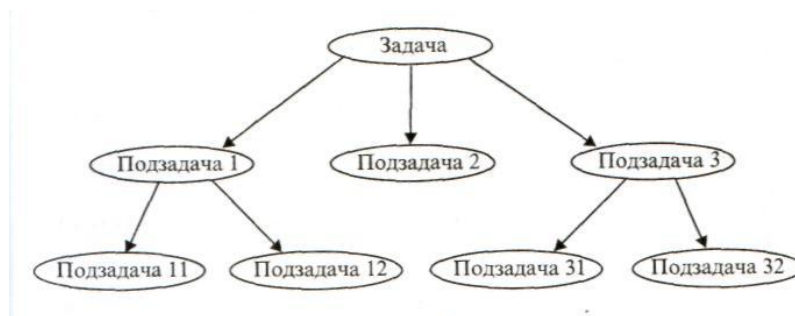
- а) Вложенные циклы-пока.
- б) Следование ветвления и цикла-до.
- в) Вложенные циклы. Внешний – цикл-пока, внутренний – цикл-до.

ТЕМА 4. ВСПОМОГАТЕЛЬНЫЕ АЛГОРИТМЫ И ПРОЦЕДУРЫ

Письменный опрос №2.

1. Какие алгоритмы называют вспомогательными?
2. какое количество вспомогательных алгоритмов может присутствовать в основном алгоритме?
3. Можно ли вспомогательные алгоритмы, написанные для решения данной задачи, использовать при решении других задач, где их применение было бы целесообразно?
4. Какие параметры называют формальными? фактическими?
5. Какое соответствие должно соблюдаться между формальными и фактическими параметрами?
6. Может ли фактических параметров процедуры (функции) быть больше, чем формальных? А меньше?
7. Существуют ли подпрограммы без параметров?
8. Существуют ли ограничения на число параметров подпрограмм? Если нет, то чем же всё-таки ограничивается это количество в Turbo Pascal?
9. В каком разделе объявляются и реализуются подпрограммы в Turbo Pascal?
10. Какие виды формальных параметров существуют? Чем они отличаются друг от друга?
11. В чём состоит отличие процедур и функций?
12. В каких случаях целесообразно использовать функции?
13. Почему, если в функции используются параметры-переменные, необходимо преобразовать её в процедуру?

14. Какого типа может быть значение функции?
15. Расскажите о методе последовательной детализации при разработке программ.
16. Какие подпрограммы называют рекурсивными?
17. Что такое граничное условие при организации рекурсивной подпрограммы?
18. Укажите номера верных предложений:
 - а) Вспомогательные алгоритмы – алгоритмы, решающие одну и ту же задачу
 - б) Кодирование – составление текста программы на языке программирования.
 - в) Эквивалентные алгоритмы – алгоритмы решения подзадач
 - г) Рекурсивный метод – сведение задачи к самой себе.
19. На рисунке показан: (выберите правильный ответ)



- а) Сборочный метод.
- б) Эвристический метод.
- в) Рекурсивный метод.
- г) Метод последовательной детализации.
- д) Метод сортировки.

ТЕМА 5. МАШИНА ПОСТА

Тест №3

Критерии оценки

Предлагаемый пакет оценочных материалов включает два равнозначных комплекта, состоящих из семи заданий.

Все задания репродуктивного характера, предполагают воспроизведение теоретического материала.

Процент правильных ответов	Оценка	Количество правильных ответов
90% – 100%	«Отлично»	12 - 11
75% – 89%	«Хорошо»	10 – 9
50% – 74%	«Удовлетворительно»	8 – 6
Менее 50%	«Неудовлетворительно»	Менее 6

1. Из чего состоит машина Поста?

- а) из ленты
- б) каретки
- в) Из ленты и каретки

2. Лента в машине Поста...

- а) конечна
- б) бесконечна

3. Как нумеруются секции на ленте машины Поста?

- а) не нумеруются
- б) относительно каретки
- в) жестко пронумерованы

4. Что записывается в секции на ленте в машине Поста?

- а) 1 или 0
- б) Ничего или V
- в) Ничего, 1,

5. В машине Поста. Информация о том, какие секции пусты, а какие отмечены, образует .

- а) состояние ленты
- б) число

6. Как может двигаться каретка в машине Поста?

- а) Влево
- б) Вправо
- в) Влево и вправо
- г) Никак

7. Сколько секций обзрывает каретка за один раз в машине Поста?

- а) Текущую и следующую
- б) Текущую
- в) следующую

8. Информация о том, какие секции пусты, а какие отмечены и где стоит каретка в машине Поста, ...

- а) Образует состояние ленты
- б) Образует состояние машины Поста
- в) Позволяет считать число с ленты

9. Сколько команд у машины Поста?

- а) 5
- б) 6
- в) 7

10. Как называется число, стоящее в конце команды машины Поста?

- а) Пересылкой
 - б) Отсылкой
 - в) Индексом
- Постфиксом

11. Чему равна длина программы машины Поста? - Количеству ...

- 1) Отсылок
- 2) Команд
- 3) Различных команд

12. Машина Поста

- 1) универсальное устройство, использующее языки программирования высокого уровня
- 2) универсальный исполнитель обработки любых символьных последовательностей в любом алфавите
- 3) работает с двоичным алфавитом
- 4) машина Тьюринга является частным случаем машины Поста

ТЕМА 6. МАШИНА ТЬЮРИНГА

Тест №4

Критерии оценки

Предлагаемый пакет оценочных материалов включает 20 заданий.

Все задания репродуктивного характера, предполагают воспроизведение теоретического материала.

Процент правильных ответов	Оценка	Количество правильных ответов
90% – 100%	«Отлично»	20 - 18
75% – 89%	«Хорошо»	17 – 15
50% – 74%	«Удовлетворительно»	14 – 10
Менее 50%	«Неудовлетворительно»	Менее 10

- В машине Тьюринга предписание L для лентопротяжного механизма означает:
 - Переместить ленту вправо
 - Переместить ленту влево
 - Остановить машину
 - Занести в ячейку символ
- В машине Тьюринга предписание R для лентопротяжного механизма означает:
 - Переместить ленту вправо
 - Переместить ленту влево
 - Остановить машину
 - Занести в ячейку символ
- В машине Тьюринга предписание S для лентопротяжного механизма означает:
 - Переместить ленту вправо
 - Переместить ленту влево
 - Остановить машину
 - Занести в ячейку символ
- В машине Тьюринга рабочий алфавит:
 - $A = \{a_{40} 0, b_{40} 1, c_{40} 2, \dots, w_{40} t\}$;
 - $A = \{a_{40} 0, a_{40} 1, a_{40} 2, \dots, a_{40} t\}$;
 - $A = \{a_{40} 0, a_{41} 0, a_{42} 0, \dots, a_{4t} 0\}$;
 - $A = \{a_{10} 0, a_{20} 0, a_{30} 0, \dots, a_{90} 0\}$
- В машине Тьюринга состояниями являются:
 - $\{a_{40} 0, a_{40} 1, a_{402}, \dots, a_{40} t\}$;
 - $\{q_{41}, q_{42}, q_{43}, \dots, q_{4s}\}$;
 - $\{q_{41}, q_{42}, q_{43}, \dots, q_{4s}, a_{40} 0, a_{40} 1, a_{40} 2, \dots, a_{40} t\}$;
 - $\{q_{40}, q_{41}, q_{42}, \dots, q_{4s}\}$.
- Результат применения команды машины Тьюринга $q_1 1 \rightarrow q_2 a_0 Lk$ машинному слову $a_0 1 * q_1 1 a_0$
 - $a_0 1 * q_2 1 a_0$
 - $a_0 1 * 1 q_2 a_0$
 - $a_0 1 q_2 * a_0$
 - $a_0 1 * q_1 1 a_0$
- Результат применения команды машины Тьюринга $q_1 1 \rightarrow q_2 a_0 Lk$ машинному слову $a_0 1 * q_1 1 a_0$
 - $a_0 1 * a_0 q_1 a_0$
 - $a_0 1 * a_0 q_2 a_0$
 - $a_0 1 q_2 * a_0$
 - $a_0 1 q_1 * a_0$
- Результат применения команды машины Тьюринга $q_1 1 \rightarrow q_2 a_0 Ck$ машинному слову $a_0 1 * q_1 1 a_0$

- 1) $a_0 1 * q_2 a_0$ 2) $a_0 1 * q_1 a_0$
 3) $a_0 1 q_2 * a_0$ 4) $a_0 1 q_1 * a_0$

9. Конфигурация машины Тьюринга, соответствующая начальному стандартному положению

- 1) $a_0 1 * q_1 a_0$ 2) $a_0 1 * q_0 a_0$
 3) $a_0 1 q_1 * a_0$ 4) $a_0 1 q_0 * a_0$

10. Конфигурация машины Тьюринга, соответствующая заключительному стандартному положению

- 1) $a_0 1 * q_1 a_0$ 2) $a_0 1 * q_0 a_0$
 3) $a_0 1 q_1 * a_0$ 4) $a_0 1 q_0 * a_0$

11. Машина Тьюринга задана функциональной схемой

$$q_1 0 \rightarrow q_1 1 \Pi$$

$$q_1 a_0 \rightarrow q_0 0 C$$

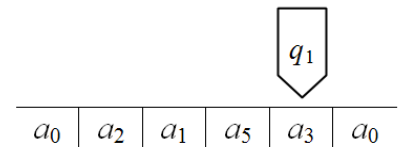
В какое слово переработает машина слово $\alpha = 01110$, исходя из стандартного начального положения

- 1) 0110 2) 011110
 2) 01111 4) 011100

12. Установите соответствие

- | | |
|------------------------|----------|
| a) пустая буква | 1) q_1 |
| b) начальное состояние | 2) q_0 |
| c) стоп-состояние | 3) a_1 |
| | 4) a_0 |

13. Конфигурация (машинное слово), соответствующая изображению



- 1) $a_0 a_2 a_1 a_5 a_3 a_0$
 2) $a_0 a_2 a_1 a_5 q_1 a_3 a_0$
 3) $a_0 a_2 a_1 a_5 a_3 q a_0$
 4) $q_1 a_0 a_2 a_1 a_5 a_3 a_0$

14. Если значение вычислимой по Тьюрингу функции $f(x_1, \dots, x_n)$ не определено, то ...

- 1) Машина останавливается через конечное число шагов, на ленте записано пустое слово
 2) Машина останавливается через конечное число шагов, на ленте записано исходное слово
 3) Машина останавливается через конечное число шагов, на ленте записано слово «error»
 4) Машина работает бесконечно

15. Функция $f(x_1, x_2)$ является вычислимой по Тьюрингу. Для вычисления значения $f(1,3)$ начальная конфигурация имеет вид

- 1) 0101110
 2) 010111 q_1 0
 3) 1*111
 4) 1*11 q_1 1

16. Функция $f(x)$ является вычислимой по Тьюрингу. Машина Тьюринга, вычисляющая ее значения, задана функциональной схемой

$$\begin{aligned} q_1 0 &\rightarrow q_1 1П \\ q_1 a_0 &\rightarrow q_0 0С \end{aligned}$$

Формульное выражение функции $f(x)$ имеет вид

- 1) $f(x) = x$
- 2) $f(x) = x + 1$
- 3) $f(x) = x + 2$
- 4) $f(x) = x - 1$

17. Функция $f(x)$ является вычислимой по Тьюрингу. Машина Тьюринга, вычисляющая ее значения, задана функциональной схемой

$$q_1 0 \rightarrow q_1 1П, \quad q_1 a_0 \rightarrow q_2 1П, \quad q_2 a_0 \rightarrow q_0 0С$$

Значение функции $f(2)$ равно ...

18. Функция $f(x)$ является вычислимой по Тьюрингу. Машина Тьюринга, вычисляющая ее значения, задана функциональной схемой

$$\begin{aligned} q_1 0 &\rightarrow q_1 0Л, & q_1 a_0 &\rightarrow q_2 a_0 П, & q_1 1 &\rightarrow q_3 0 П, \\ q_2 0 &\rightarrow q_0 0 П, & q_3 0 &\rightarrow q_0 a_0 Л \end{aligned}$$

Значение функции $f(2)$ равно ...

19. Функция $f(x)$ является вычислимой по Тьюрингу. Машина Тьюринга, вычисляющая ее значения, задана функциональной схемой

$$\begin{aligned} q_1 0 &\rightarrow q_1 0Л, & q_1 a_0 &\rightarrow q_2 a_0 П, & q_1 1 &\rightarrow q_3 0 П, \\ q_2 0 &\rightarrow q_0 0 П, & q_3 0 &\rightarrow q_0 a_0 Л \end{aligned}$$

Значение функции $f(0)$ равно ...

20. Функция $f(x)$ является вычислимой по Тьюрингу. Машина Тьюринга, вычисляющая ее значения, задана функциональной схемой

$$\begin{aligned} q_1 0 &\rightarrow q_1 0Л, & q_1 a_0 &\rightarrow q_2 a_0 П, & q_1 1 &\rightarrow q_3 0 П, \\ q_2 0 &\rightarrow q_0 0 П, & q_3 0 &\rightarrow q_0 a_0 Л \end{aligned}$$

ТЕМА 7. АЛГОРИТМЫ МАРКОВА

Тест №5

Критерии оценки

Предлагаемый пакет оценочных материалов включает 17 заданий.

Все задания репродуктивного характера, предполагают воспроизведение теоретического материала.

Процент правильных ответов	Оценка	Количество правильных ответов
90% – 100%	«Отлично»	17 – 15
75% – 89%	«Хорошо»	14 – 12
50% – 74%	«Удовлетворительно»	11 – 8
Менее 50%	«Неудовлетворительно»	Менее 8

1 вариант

1. Результат марковской подстановки $21 \rightarrow 3$ в слово 521421
 - a) 52143
 - b) 5343
 - c) 531421
 - d) 533433
2. Марковская подстановка $21 \rightarrow 3$ не применима к словам
 - a) 521421
 - b) 5241
 - c) 21
 - d) 12
3. Результат марковской подстановки $21 \rightarrow \Lambda$ в слово 521421
 - a) 54
 - b) 21521421
 - c) 5421
 - d) 52142121
4. Слово 21 является подсловом слова
 - a) 521421
 - b) 5241
 - c) 521
 - d) 2541
5. Процесс работы нормального алгоритма считается завершенным, если на данном шаге
 - a) Применена последняя формула в списке формул марковских подстановок, задающих данный алгоритм
 - b) Применена заключительная формула подстановки
 - c) Ни одна подстановка схемы не подходит
 - d) Понятно, что процесс подстановок не сможет остановиться
6. Нормальный алгоритм не применим к исходным данным, если на данном шаге
 - a) Применена последняя формула в списке формул марковских подстановок, задающих данный алгоритм
 - b) Применена заключительная формула подстановки
 - c) Ни одна подстановка схемы не подходит
 - d) Понятно, что процесс подстановок не сможет остановиться
7. Каким будет результат применения нормального алгоритма $\alpha\beta \rightarrow \Lambda$, $\beta\alpha \rightarrow \alpha\beta$ к слову $R = \beta\beta\alpha\alpha\beta\alpha\beta$
8. Результат применения нормального алгоритма $ab \rightarrow bd$, $db \rightarrow ba$, $bba \rightarrow abb$, $c \rightarrow \Lambda$ к слову $R = abbc$
 - a) Алгоритм не применим к этому слову
 - b) bb
 - c) aa
 - d) cc
9. В алгоритме Маркова ассоциативным исчислением называется:
 - a) совокупность всех слов в данном алфавите;
 - b) совокупность всех допустимых систем подстановок;
 - c) совокупность всех слов в данном алфавите вместе с допустимой системой подстановок;
 - d) когда все слова в алфавите являются смежными.
10. В ассоциативном счислении два слова называются смежными:
 - a) если одно из них может быть преобразовано в другое применением подстановок;
 - b) если одно из них может быть преобразовано в другое однократным применением допустимой подстановки;
 - c) когда существует цепочка от одного слова к другому и обратно;
 - d) когда они дедуктивны.
11. В алгоритмах Маркова дана система подстановок в алфавите $A = \{a, b, c\}$: $abc \rightarrow c$; $ba \rightarrow cb$; $ca \rightarrow ab$
 Преобразуйте с помощью этой системы слово $bacaabc$

12. В алгоритмах Маркова дана система подстановок в алфавите $A = \{a, b, c\}$: $cb \rightarrow abc$; $bac \rightarrow ac$; $cab \rightarrow b$

Преобразуйте с помощью этой системы слово $bcabacab$

13. Способ композиции нормальных алгоритмов будет суперпозицией, если:
- выходное слово первого алгоритма является входным для второго;
 - существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A , B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - существует алгоритм C , являющийся суперпозицией алгоритмов A и B такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
14. Способ композиции нормальных алгоритмов будет объединением, если:
- выходное слово первого алгоритма является входным для второго;
 - существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A , B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - существует алгоритм C , являющийся суперпозицией алгоритмов A и B такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
15. Способ композиции нормальных алгоритмов будет разветвлением, если:
- выходное слово первого алгоритма является входным для второго;
 - существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A , B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - существует алгоритм C , являющийся суперпозицией алгоритмов A и B , такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
16. Способ композиции нормальных алгоритмов будет итерацией, если:
- выходное слово первого алгоритма является входным для второго;
 - существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A , B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - существует алгоритм C , являющийся суперпозицией алгоритмов A и B , такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не

получится слово, преобразуемое алгоритмом B .

17. Дать подробное решение и описание алгоритма решения задачи:

- 1) $A = \{f, h, p\}$ в слове P заменить все пары ph на f
- 2) $A = \{a, b, c\}$ Приписать словобас слева к слову P

2 вариант

1. Каким будет результат применения нормального алгоритма $\alpha\beta \rightarrow \Lambda$, $\beta\alpha \rightarrow \alpha\beta$ к слову $R = \beta\beta\alpha\alpha\beta\alpha\beta$
2. Слово 21 является подсловом слова
 - a) 521421
 - b) 5241
 - c) 521
 - d) 2541
3. Нормальный алгоритм не применим к исходным данным, если на данном шаге
 - a) Применена последняя формула в списке формул марковских подстановок, задающих данный алгоритм
 - b) Применена заключительная формула подстановки
 - c) Ни одна подстановка схемы не подходит
 - d) Понятно, что процесс подстановок не сможет остановиться
4. В алгоритмах Маркова дана система подстановок в алфавите
 $A = \{a, b, c\}: cb \rightarrow abc; \quad bac \rightarrow ac; \quad cab \rightarrow b$
Преобразуйте с помощью этой системы слово $bcabacab$
5. Способ композиции нормальных алгоритмов будет итерацией, если:
 - a) выходное слово первого алгоритма является входным для второго;
 - b) существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - c) алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - d) существует алгоритм C , являющийся суперпозицией алгоритмов A и B , такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
6. Способ композиции нормальных алгоритмов будет объединением, если:
 - a) выходное слово первого алгоритма является входным для второго;
 - b) существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - c) алгоритм B будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - d) существует алгоритм C , являющийся суперпозицией алгоритмов A и B такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
7. Способ композиции нормальных алгоритмов будет суперпозицией, если:
 - a) выходное слово первого алгоритма является входным для второго;
 - b) существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - c) алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения

- алгоритмов A , B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
- d) существует алгоритм C , являющийся суперпозицией алгоритмов A и B такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
8. Способ композиции нормальных алгоритмов будет разветвлением, если:
- выходное слово первого алгоритма является входным для второго;
 - существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B ;
 - алгоритм D будет суперпозицией трех алгоритмов A, B, C , причем область определения D является пересечением областей определения алгоритмов A, B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, если $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e — пустая строка;
 - существует алгоритм C , являющийся суперпозицией алгоритмов A и B , такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B .
9. Процесс работы нормального алгоритма считается завершённым, если на данном шаге
- Применена последняя формула в списке формул марковских подстановок, задающих данный алгоритм
 - Применена заключительная формула подстановки
 - Ни одна подстановка схемы не подходит
 - Понятно, что процесс подстановок не сможет остановиться
10. Результат марковской подстановки $21 \rightarrow 3$ в слово 521421
- 52143
 - 5343
 - 531421
 - 533433
11. Результат применения нормального алгоритма $ab \rightarrow bd, db \rightarrow ba, bba \rightarrow abb, c \rightarrow \Lambda$ к слову $R = abbc$
- Алгоритм не применим к этому слову
 - bb
 - aa
 - cc
12. В алгоритмах Маркова дана система подстановок в алфавите $A = \{a, b, c\}$: $abc \rightarrow c$; $ba \rightarrow cb$; $ca \rightarrow ab$
Преобразуйте с помощью этой системы слово $bacaaabc$
13. алгоритме Маркова ассоциативным исчислением называется:
- совокупность всех слов в данном алфавите;
 - совокупность всех допустимых систем подстановок;
 - совокупность всех слов в данном алфавите вместе с допустимой системой подстановок;
 - когда все слова в алфавите являются смежными.
14. В ассоциативном счислении два слова называются смежными:
- если одно из них может быть преобразовано в другое применением подстановок;
 - если одно из них может быть преобразовано в другое однократным применением допустимой подстановки;
 - когда существует цепочка от одного слова к другому и обратно;
 - когда они дедуктивны.

15. Марковская подстановка $21 \rightarrow 3$ не применима к словам
 a) 521421 b) 5241
 c) 21 d) 12
16. Результат марковской подстановки $21 \rightarrow \Lambda$ в слово 521421
 a) 54 b) 21521421
 c) 5421 d) 52142121
17. Дать подробное решение и описание алгоритма решения задачи:
 3) $A = \{a, b, c\}$ в слове P заменить все пары ac на b
 $A = \{p, f\}$ Приписать слово frf слева к слову S

2) Письменный опрос (примерные задания)

Письменный опрос №3.

1. В чем заключается суть структурного программирования? .
2. Приведите пример задачи, поддающейся последовательной детализации.
3. Что такое подпрограмма?
4. От чего зависит количество подпрограмм в программе?
5. В каком случае действия целесообразно оформлять в виде подпрограммы?
6. Как можно классифицировать переменные в зависимости от уровня использования переменных: в основной программе или в подпрограмме?
7. В чем различие формальных и фактических параметров?
8. Какими бывают формальные параметры подпрограммы?
9. Для чего используются параметры-значения? Изменяются ли параметры-значения после выполнения подпрограммы?
10. Для чего используются параметры-переменные? Изменяются ли параметры-переменные после выполнения подпрограммы?
11. Какое соответствие должно быть установлено между формальными и фактическими параметрами?
12. Должны ли фактические параметры соответствовать формальным по типу?
13. Должны ли фактические параметры соответствовать формальным по количеству?
14. Должны ли фактические параметры соответствовать формальным по порядку?
15. Должны ли фактические параметры соответствовать формальным по имени?
16. Перечислите принципы структурного программирования.
17. Важнейший принцип структурного программирования базируется на утверждении:
 - а) сущность формализации решаемой задачи заключается в составлении алгоритма с помощью следующих базовых структур линейной, ветвящейся, циклической;
 - б) любой алгоритм имеет дискретную структуру;
 - в) в качестве обязательного этапа создания программы выступает ее тестирование и отладка;
 - г) алгоритм любой сложности можно построить с помощью следующих базовых структур линейной, ветвящейся, циклической.
18. При структурном подходе к составлению программ могут использоваться понятия:
 - а) альтернативный выбор
 - б) цикл
 - с) подпрограмма
 - д) наследование
19. В чем суть метода последовательной детализации?
20. Что такое программирование снизу вверх; сверху вниз?
21. Когда следует применять рекурсивные алгоритмы?
22. Какие известны методы и приемы устранения "хвостовой" рекурсии?

23. Какие проблемы могут возникать при реализации рекурсивных алгоритмов на электронных вычислительных машинах?
24. В чем отличие глубины рекурсии от рекурсивного вызова?
25. Какие задачи в программировании можно назвать, где применение рекурсии оправдано?

3) Типовая практическая работа

Практическая работа 1. Разработка линейных алгоритмов.

Учебная цель: **Формирование представлений о линейных алгоритмах, навыков записи алгоритмов с помощью блок-схем и умений устанавливать соответствие между командами алгоритма, записанного словесно, и элементами блок-схемы**

Учебные задачи:

1. Овладеть практическими навыками разработки линейных алгоритмов.
2. Овладеть практическими навыками работы в MS VISIO 2007: изучить интерфейс, изучить основные инструменты создания и оформления блок-схем.

Образовательные результаты, заявленные во ФГОС третьего поколения:

Студент должен

уметь:

- разрабатывать алгоритмы для конкретных задач;

знать:

- основные модели алгоритмов;
- методы построения алгоритмов.

Задачи практической работы:

1. Повторить теоретический материал по теме практической работы.
2. Ответить на вопросы для закрепления теоретического материала.
3. Разработать линейные блок-схемы.
4. Оформить отчет.

Краткие теоретические и учебно-методические материалы по теме практической работы

Линейные алгоритмы - это такие алгоритмы, в которых действия совершаются одно за другим, в строго определенном порядке

Рассмотрим пример составления линейного вычислительного алгоритма.

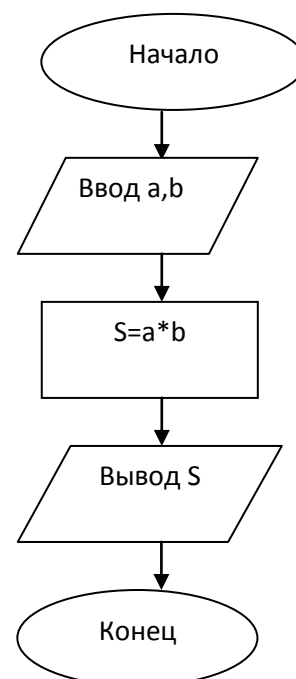
Пример 1: Вычислить площадь прямоугольника по заданной длине и ширине:

1. Ввести a, b
2. Вычислить площадь S по формуле $a*b$
3. Вывести полученный результат на экран.
4. Закончить выполнение алгоритма.

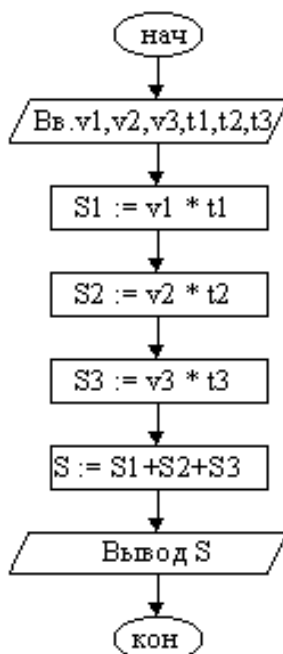
Это словесно – формульная форма записи алгоритма. Посмотрим, как выглядит данная задача, записанная с помощью графических блоков.

Вычислительных действий в блок-схеме может быть и несколько, в зависимости от условия задачи и от оформления.

Пример 2. Пешеход шел по пересеченной местности. Его скорость движения по равнине v_1 км/ч, в гору — v_2 км/ч и под гору — v_3 км/ч. Время движения соответственно t_1, t_2 и t_3 ч. Какой полный путь прошел пешеход?



1. Ввести $v_1, v_2, v_3, t_1, t_2, t_3$.
2. $S_1 := v_1 * t_1$.
3. $S_2 := v_2 * t_2$.
4. $S_3 := v_3 * t_3$.
5. $S := S_1 + S_2 + S_3$.
6. Вывести значение S .
7. Конец.



Интерфейс окна приложения Visio

При открытии MS Visio по умолчанию в главном окне два дополнительных. Справа – это окно Область задач и слева – окно Библиотеки шаблонов. С помощью шаблонов создаются всевозможные документы деловой графики – рисунки, схемы и диаграммы.

Синяя полоса в верхней части окна называется Строка заголовка. В ней написано имя открытого приложения, под ней – Панель меню.

СОЗДАНИЕ НОВОГО ДОКУМЕНТА С ПОМОЩЬЮ ШАБЛОНОВ

Для того, чтобы начать работу в программе MS Visio нужно либо создать новый документ, либо открыть уже созданный документ и продолжить работу в нем. Документ в MS Visio - это файл, который содержит листы с созданными пользователем изображениями. Рассмотрим сначала какие существуют способы создания нового документа. В меню Файл, выбираем подменю Новый, затем необходимо выбрать необходимый шаблон.

НАСТРОЙКА СТРАНИЦЫ

Размер бумаги. Для выбора предложены размеры бумаги, которые поддерживаются текущим принтером, а также стандартные размеры. Если установить флажок Как в принтере на вкладке Размер страницы, изменения размера бумаги будут применяться также к странице документа.

В меню Файл выберите команду Предварительный просмотр, чтобы проверить соответствие размеров страницы документа размерам бумаги для печати.

Ориентация бумаги. Установите один из этих флажков, чтобы задать книжную или альбомную ориентацию страницы документа. Если на вкладке Размер страницы

установлен флажок Как в принтере, изменения ориентации бумаги будут также применены к странице документа.

В меню Файл выберите команду Предварительный просмотр, чтобы проверить соответствие размеров страницы документа размерам бумаги для печати.


Настройка. При нажатии этой кнопки открывается диалоговое окно Настройка печати, в котором можно выбрать такие параметры печати, как поля, центрирование, принтер и источник бумаги.

Сетка. Установите этот флажок, чтобы вывести на печать линии сетки, отображаемые в окне документа.

Просмотр. В этой части окна отображается эскиз страницы, позволяющий проверить соотношение размеров текущей страницы документа и бумаги для печати.

Можно задать одинаковые размеры страницы документа и бумаги для печати. Для этого на вкладке Размер страницы установите флажок Как в принтере.

ДОБАВЛЕНИЕ НОВОЙ СТРАНИЦЫ.

1. В левом нижнем углу окна документа щелкните правой кнопкой мыши вкладку страницы , а затем выберите в контекстном меню команду Вставить страницу.

2. Перейдите на вкладку Свойства страницы и введите имя для страницы или воспользуйтесь именем по умолчанию.

3. Для изменения масштаба или размера новой страницы перейдите на вкладку Масштаб документа или Размер страницы.

СОЗДАНИЕ ПРОСТОЙ БЛОК-СХЕМЫ

С помощью блок-схем можно документировать процедуры, анализировать процессы, обозначать рабочий или информационный процессы, затраты на отслеживание, эффективность и т. д.

1. В меню Файл последовательно выберите команды Создать, Бизнес или Блоксхема, а затем — команду Простая блок-схема.

2. Для каждого шага документируемого процесса перетащите в документ фигуру блок-схемы.

3. Соедините фигуры блок-схемы.

a. Перетащите фигуру из набора элементов на страницу документа и расположите ее вблизи другой фигуры.

b. Не отпуская кнопку мыши, переместите указатель на один из светло-синих треугольных маркеров. Цвет маркера изменится на темно-синий.

c. Отпустите кнопку мыши. Фигура будет размещена на странице, и к обеим фигурам будет добавлена и приклеена соединительная линия.

4. Для добавления текста в фигуру выделите ее, а затем введите текст. Закончив ввод, щелкните за пределами текстового блока.

5. Для отображения последовательности шагов процесса фигуры в блок-схеме можно пронумеровать.

a. В блок-схеме выделите фигуры, которым нужно присвоить номера.

В меню Сервис последовательно выберите команды Надстройки, Дополнительные решения Visio, а затем — команду Нумерация фигур.

c. На вкладке Общие в группе Операция установите переключатель в положение Автонумерация.

d. В группе Применить к установите переключатель в положение Выбранные фигуры, а затем нажмите кнопку ОК Совет. Для автоматической нумерации новых фигуры блок-схемы по мере их перетаскивания на страницу, в диалоговом окне Нумерация фигур установите флажок Продолжать нумерацию фигур при перетаскивании на страницу

Задания для практического занятия:

1. Рассмотреть примеры 1-2.
2. Разработать словесно -формульный алгоритм по индивидуальному заданию.
3. Разработать блок-схему алгоритма решения задачи средствами MS Visio.
4. Оформить отчет в MS Word.

Индивидуальные задания

Вариант 1.

1. Дана сторона квадрата. Найти его периметр.
2. Дан радиус окружности. Найти длину окружности и площадь круга.
3. Известны количество жителей в государстве и площадь его территории. Определить плотность населения в этом государстве.
4. Даны катеты прямоугольного треугольника. Найти его периметр.
5. Треугольник задан координатами своих вершин. Найти периметр и площадь треугольника.
6. Дано двузначное число. Найти:
 - а) число десятков в нем;
 - б) число единиц в нем;
 - в) сумму его цифр;
 - г) произведение его цифр.

Вариант 2

1. Дан радиус окружности. Найти ее диаметр.
2. Даны два целых числа. Найти их среднее арифметическое.
3. Даны катеты прямоугольного треугольника. Найти его гипотенузу.
4. Даны два числа. Найти их сумму, разность, произведение, а также частное от деления первого числа на второе.
5. Даны длины сторон прямоугольного параллелепипеда. Найти его объем и площадь боковой поверхности.
6. Дано трехзначное число. Найти:
 - а) число единиц в нем;
 - б) число десятков в нем;
 - в) сумму его цифр;
 - г) произведение его цифр.

Вариант 3

1. Дана длина ребра куба. Найти объем куба и площадь его боковой поверхности.
2. Даны два целых числа. Найти их среднее геометрическое.
3. Найти площадь кольца по заданным внешнему и внутреннему радиусам.
4. Даны два числа. Найти среднее арифметическое и среднее геометрическое их модулей.
5. Даны длины сторон прямоугольника. Найти его периметр и длину диагонали.
6. Дано двузначное число. Получить число, образованное при перестановке цифр заданного числа.

Методика анализа результатов, полученных в ходе практической работы

1. В результате выполнения заданий у Вас должны быть алгоритмы описанные словесно-формульным способом и в виде блок-схемы, разработанной в MS Visio по индивидуальным заданиям.

Порядок выполнения отчета по практической работе

1. Разработать программу по индивидуальному заданию.
2. Предоставить преподавателю отчет в MS Word.

Пример отчета

Практическая работа. Разработка линейных алгоритмов.

ЦЕЛЬ РАБОТЫ: Формирование представлений о линейных алгоритмах, отработка навыков записи алгоритмов с помощью блок-схем и умений устанавливать соответствие между командами алгоритма, записанного словесно, и элементами блок-схемы

Индивидуальные задания

Вариант 1.

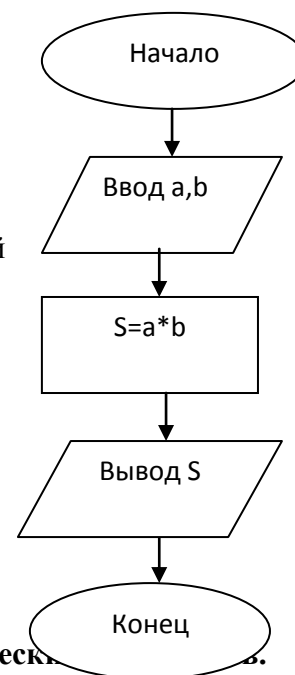
1. Вычислить площадь прямоугольника по заданной длине и ширине;
2. Дан радиус окружности. Найти длину окружности и площадь круга.
3. Известны количество жителей в государстве и площадь его территории. Определить плотность населения в этом государстве.
4. Даны катеты прямоугольного треугольника. Найти его периметр.
5. Треугольник задан координатами своих вершин. Найти периметр и площадь треугольника.
6. Дано двузначное число. Найти:
 - а) число десятков в нем;
 - б) число единиц в нем;
 - в) сумму его цифр;
 - г) произведение его цифр.

Задание 1: Вычислить площадь прямоугольника по заданной длине и ширине:

Словесно -формульный алгоритм

Блок-схема:

- | |
|---|
| <ol style="list-style-type: none">1. Ввести a, b2. Вычислить площадь S по формуле $a*b$3. Вывести полученный результат на экран.4. Закончить выполнение алгоритма |
|---|



Практическая работа: Разработка разветвляющихся и циклических алгоритмов.

Учебная цель: сформировать умение разрабатывать разветвляющиеся и циклические алгоритмы

Учебные задачи:

1. Овладеть практическими навыками разработки блок-схем для разветвляющихся алгоритмов.
2. Овладеть практическими навыками разработки блок-схем для циклических алгоритмов.

Образовательные результаты, заявленные во ФГОС третьего поколения:

Студент должен

уметь:

- разрабатывать алгоритмы для конкретных задач;

знать:

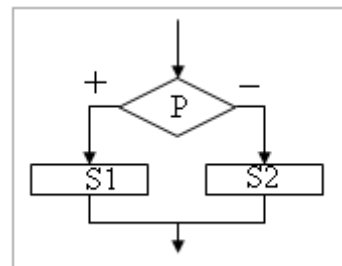
- основные модели алгоритмов;
- методы построения алгоритмов.

Задачи практической работы:

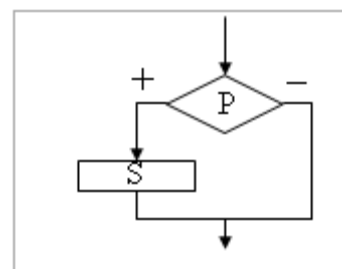
1. Повторить теоретический материал по теме практической работы.
2. Ответить на вопросы для закрепления теоретического материала.
3. Разработать блок-схемы.
4. Оформить отчет.

Краткие теоретические и учебно-методические материалы по теме практической работы

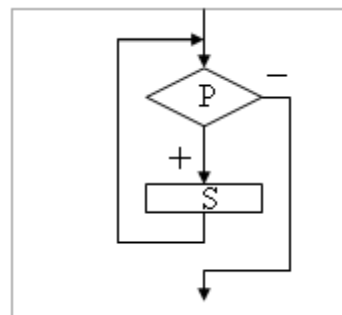
Команда ветвления - это составная команда алгоритма, в которой в зависимости от условия P выполняется или одно $S1$, или другое $S2$ действие. Из команд следования и команд ветвления составляются разветвляющиеся алгоритмы (алгоритмы ветвления). Примером разветвляющегося алгоритма будет нахождение большего из двух чисел, введенных с клавиатуры.



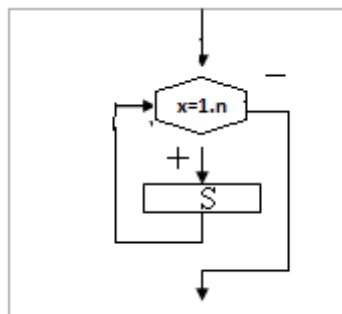
Команда ветвления может быть полной и неполной формы. Неполная форма команды ветвления используется тогда, когда необходимо выполнять действие S только в случае соблюдения условия P . Если условие P не соблюдается, то команда ветвления завершает свою работу без выполнения действия. Примером команды ветвления неполной формы будет уменьшение в два раза только четного числа.



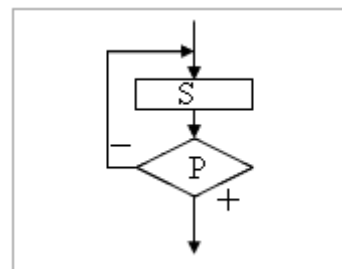
Команда повторения - это составная команда алгоритма, в которой в зависимости от условия P возможно многократное выполнение действия S . Из команд следования и команд повторения составляются циклические алгоритмы (алгоритмы повторения). На рисунке представлена команда повторения с предусловием. Называется она так потому, что вначале проверяется условие, а уже затем выполняется действие. Причем действие выполняется, пока условие соблюдается. Пример циклического алгоритма может быть следующий. Пока с клавиатуры вводятся положительные числа, алгоритм выполняет нахождение их суммы.



Команда повторения с предусловием не является единственно возможной. Разновидностью команды повторения с предусловием является **команда повторения с параметром**. Она используется тогда, когда известно количество повторений действия. В блок-схеме команды повторения с параметром условие записывается не в ромбе, а в шестиугольнике. Примером циклического алгоритма с параметром будет нахождение суммы первых 20 натуральных чисел.



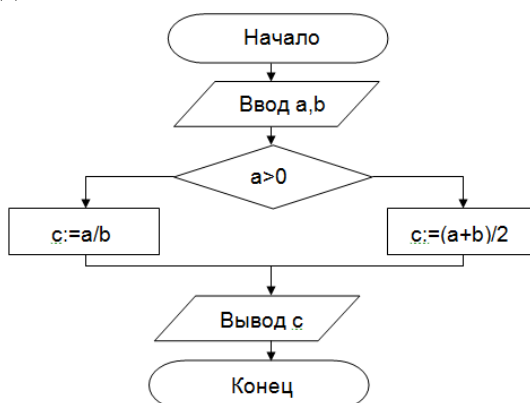
В команде повторения с постусловием вначале выполняется действие S и лишь затем, проверяется условие P . Причем действие повторяется до тех пор, пока условие не соблюдается. Примером команды повторения с постусловием будет уменьшение положительного числа до тех пор, пока оно неотрицательное. Как только число становится отрицательным, команда повторения заканчивает свою работу.



С помощью соединения только этих элементарных конструкций (последовательно или вложением) можно "собрать" алгоритм любой степени сложности.

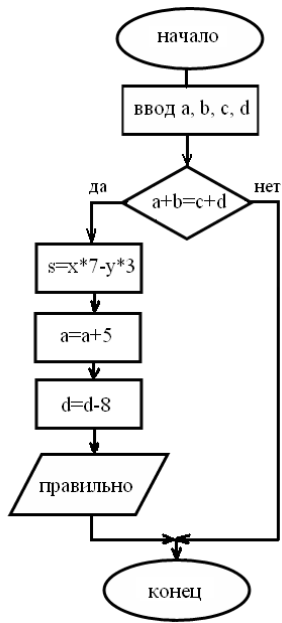
Пример 1. Определить среднее арифметическое двух чисел, если a положительное и частное (a/b) в противном случае.

Запись алгоритма в виде блок-схемы



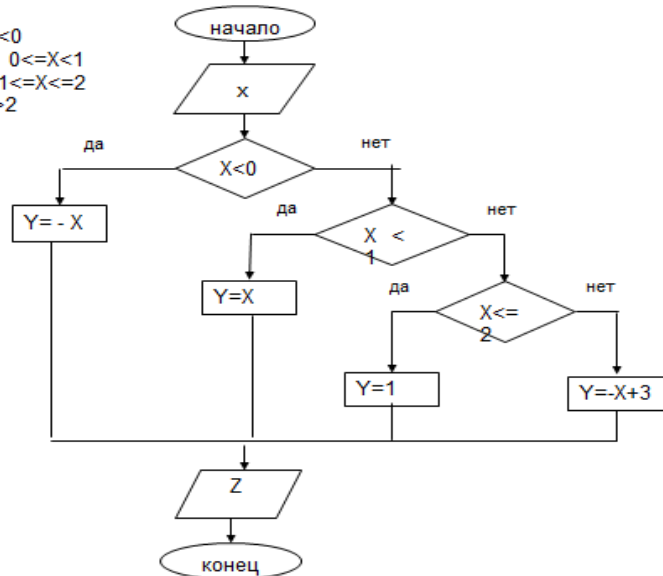
Пример 2. Составить блок-схему решения задачи:

Если $a+b=c+d$ то сделать следующее: напечатать сообщение "Правильно", вычислить $s=x*7-y*3$, сложить a с пятью, вычесть из d восемь.



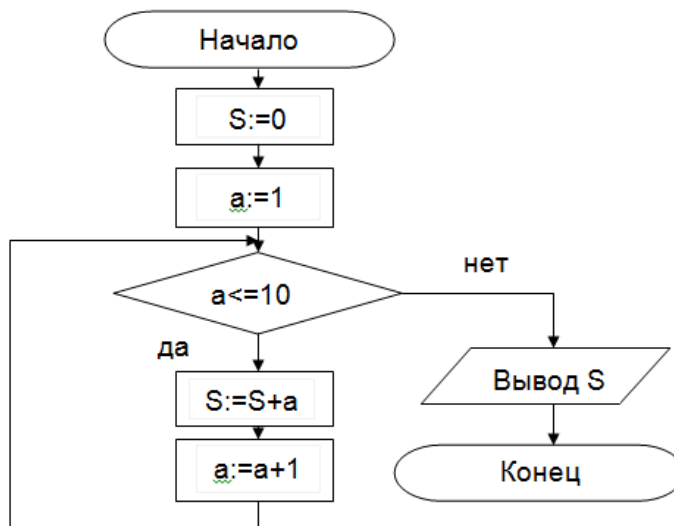
Пример 3. По заданной функции нарисовать блок-схему.

$$Y = \begin{cases} -X, & \text{если } X < 0 \\ X, & \text{если } 0 \leq X < 1 \\ 1, & \text{если } 1 \leq X \leq 2 \\ -X + 3, & \text{если } X > 2 \end{cases}$$



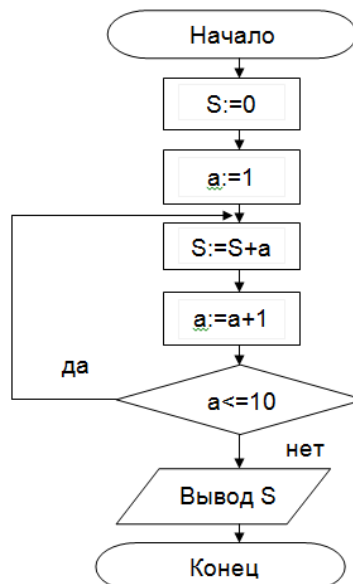
Пример 4. Составить алгоритм нахождения суммы целых чисел в диапазоне от 1 до 10.

Запись алгоритма в виде блок-схемы:



В алгоритме с постусловием сначала выполняется тело цикла, а затем проверяется условие окончания цикла. Решение задачи нахождения суммы первых десяти целых чисел в данном случае будет выглядеть следующим образом:

Запись алгоритма в виде блок-схемы:



Вопросы для закрепления теоретического материала к практическому занятию:

1. Что такое алгоритм?
2. Свойства алгоритма.
3. Способы записи алгоритма.
4. Основные элементы блок-схемы.
5. Виды алгоритмов.
6. Отличительные особенности алгоритмов с предусловием и постусловием.

Задания для практического занятия:

1. Рассмотреть примеры.
2. Составить алгоритм решения задачи в виде блок-схемы по индивидуальному заданию
3. Оформить отчет

Индивидуальные задания

Вариант 1.

1. Задано число x . Вычислить функцию s .

$$s = -1, \text{ если } x < 0$$

$$s = 0, \text{ если } x = 0$$

$$s = 1, \text{ если } x > 0$$

1. Составить алгоритм нахождения суммы S и произведения p четных чисел от 1 до n .

Вариант 2.

1. Составить блок-схему решения задачи. Заданы два числа k и m . Переменной n присвоить значение 1, если $k < m$, 0, если $k = m$, и -1, если $k > m$.

2. Составить алгоритм нахождения суммы S и произведения p нечетных чисел от 1 до n .

Вариант 3.

1. Составить блок-схему вычисления функции:

$$X, \text{ если } X < 2$$

$$Y = 2, \text{ если } 2 \leq X \leq 3$$

$$-X + 5, \text{ если } X > 3$$

2. Составить алгоритм нахождения суммы S и произведения p чисел, кратных 3 от 1 до n .

Методика анализа результатов, полученных в ходе практической работы

1. В результате выполнения заданий у Вас должны быть разработаны алгоритмы средствами MS Visio.

Порядок выполнения отчета по практической работе

1. Разработать блок-схемы по индивидуальному заданию.
2. Предоставить преподавателю отчет.

Содержание отчета

1. Цель работы и задание.
2. Условие задачи.
3. Алгоритм, написанный с помощью блок-схемы.

Пример отчета

Практическая работа. Разработка разветвляющихся и циклических алгоритмов.

Учебная цель: сформировать умение разрабатывать разветвляющиеся и циклические алгоритмы

4) Самостоятельная работа

ЗАДАЧА. Разработать алгоритм вычисления наибольшего числа из двух чисел x и y .

Этап 1. Математическое описание решения задачи.

Из курса математики известно, если $x > y$, то наибольшее число x , если $x < y$, то наибольшее число y , если $x = y$, то число x равно числу y .

Этап 2. Определение входных и выходных данных.

Входными данными являются значения чисел x и y . Выходным данными являются:

- наибольшее число
- любое из чисел, если числа равны

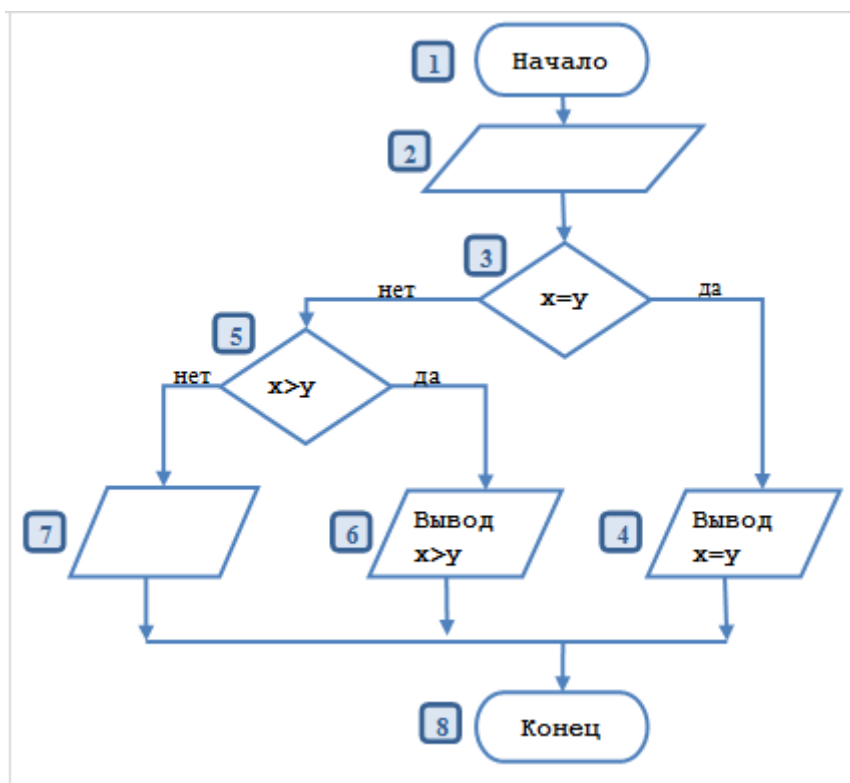
Для решения задачи нам необходимо знать значения x и y .

Этап 3. Разработка алгоритма решения задачи.

Словесное описание алгоритма

1. Начало алгоритма.
2. Ввод значений x и y .
3. Сравниваем x и y .
Если $x = y$, то переход к шагу 4, иначе к шагу 5.
4. Вывод информации: числа x и y равны. Переход к шагу 8.
5. Сравниваем x и y .
Если $x > y$, то переход к шагу 6, иначе к шагу 7.
6. Вывод информации: число x больше y . Переход к шагу 8.
7. Вывод информации: число y больше x . Переход к шагу 8.
8. Конец алгоритма.

Запись алгоритма на языке блок-схем



ЗАДАЧА. Разработать алгоритм вычисления суммы натуральных чисел от 1 до 100.

Этап 1. Математическое описание решения задачи.

Обозначим сумму натуральных чисел через S . Тогда формула вычисления суммы натуральных чисел от 1 до 100 может быть записана так:

$$S = 1 + 2 + 3 + \dots + 97 + 98 + 99 + 100 = \sum_{i=1}^n X_i$$

где X_i – натуральное число X с номером i , который изменяется от 1 до n , $n=100$ – количество натуральных чисел.

Этап 2. Определение входных и выходных данных.

Входными данными являются натуральные числа: 1, 2, 3, 4, 5, ..., 98, 99, 100.

Выходные данные – значение суммы членов последовательности натуральных чисел.

Параметр цикла – величина, определяющая количество повторений цикла. В нашем случае i – номер натурального числа.

Подготовка цикла заключается в задании начального и конечного значений параметра цикла.

- начальное значение параметра цикла равно 1,
- конечное значение параметра цикла равно n ,
- шаг цикла равен 1.

Для корректного суммирования необходимо предварительно задать начальное значение суммы, равное 0.

Тело цикла. В теле цикла будет выполняться накопление значения суммы чисел, а также вычисляться следующее значение параметра цикла по формулам:

$$S=S+i; \quad I=I+1;$$

Условие продолжения цикла: цикл должен повторяться до тех пор, пока не будет добавлен последний член последовательности натуральных чисел, т.е. пока параметр цикла будет меньше или равен конечному значению параметра цикла.

Этап 3. Разработка алгоритма решения задачи.

Введем обозначения: S – сумма последовательности, i – значение натурального числа.

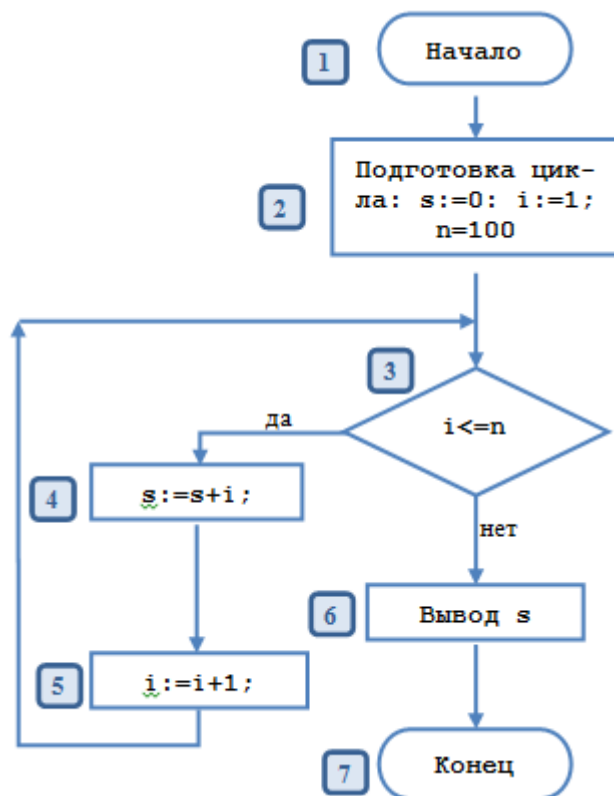
Начальное значение цикла $i=1$, конечное значение цикла $i=100$, шаг цикла 1.

Словесное описание алгоритма

1. Начало алгоритма.
2. Подготовка цикла: $S:=0$; $i:=1$; $n=100$;
3. Проверка условия. Если $i \leq n$, то перейти к шагу 4, иначе к шагу 6.
4. Накопление суммы: $S:=S+i$;
5. Вычисление следующего значения параметра цикла: $i:=i+1$;
6. Вывод информации: сумма натуральных чисел – S .
7. Конец алгоритма.

Запись алгоритма на языке блок-схем

В схеме алгоритма решения задачи цифрами указаны номера элементов алгоритма. Номера элементов соответствуют номерам шагов словесного описания алгоритма.



Типовая самостоятельная работа

Самостоятельная работа 1. Составление хронологической таблицы фундаментальных достижений в области Теории алгоритмов

Методические указания по выполнению задания:

Заполнить хронологическую таблицу фундаментальных достижений в области Теории алгоритмов

Имя (полностью), Фамилия	Даты жизни	Достижения в области Теории алгоритмов	Практическое применение данных достижений
Евклид			
Аль-Хорезми			
Г. Лейбниц			
К. Гёдель			
А. Тьюринг			
Э. Пост			
С. Клини			
А.А. Марков			
Ю.В. Матиясевич			

Рекомендации по составлению таблиц

Тезисные таблицы предпочтительны по той причине, что они не только дают впоследствии возможность восстановить содержание и главные моменты изучаемого учебного материала, выделить в нем главное, но также обеспечивают возможность определения их взаимосвязи друг с другом, или сравнения. При этом главные моменты усваиваются намного быстрее, нежели в конспектах. Кроме того, при желании эти главные моменты могут быть поставлены в виде ключевых вопросов для развернутого ответа на них своими словами. Наконец, тезисная таблица - самая простая в составлении, что немаловажно в условиях дефицита времени для полных записей студентами.

Основные правила составления таблиц

Таблица – перечень сведений, числовых данных, приведенных в определенную систему и разнесенных по графам.

Следует соблюдать следующие правила:

1. Таблица должна быть настолько полной, чтобы ее можно было понять без постоянных отсылок к основному тексту, однако она должна содержать только необходимые данные.
2. Таблица должна быть по возможности простой.
3. Таблица должна быть составлена компактно, т. е. быть небольшой по размеру и легко обозримой.
4. Общий заголовок таблицы должен кратко выражать ее основное содержание.
5. Вертикальные и горизонтальные графы таблицы (столбцы и строки) должны быть расположены в определенной логической последовательности.
6. Единицы измерения, символы и данные в таблице должны совпадать с этими элементами в тексте.
7. Одни и те же данные нельзя представлять и в таблицах, и на рисунках,
8. Данные должны быть четко структурированы.
9. Для удобной работы с цифровым материалом числа в таблицах следует расставлять в середине граф, одно под другим: единицы под единицами, запятая под запятой и г, д., четко соблюдая при этом их разрядность.
10. В таблицу можно включать примечания, в которых будут указываться источники данных, более подробное содержание показателей и другие необходимые пояснения.

Составные части таблицы

Таблица состоит из пяти основных частей:

А. Номер и название.

Б. Заголовки столбцов.

В. Боковик (содержит заголовки строк).

Г. Поле данных (содержит сведения, числовые данные).

Д. Примечания.

У каждого столбца должно быть название.

Основные части таблицы разделяют тремя горизонтальными линиями. Первая линия расположена под номером и названием таблицы, т. е. над ее головкой, вторая – под головкой, т.е. над боковиком и полем, и третья – под боковиком и полем, т. е. над примечаниями.

Каждая таблица статьи должна иметь уникальный номер и название. Таблицы нумеруют арабскими цифрами и располагают в порядке первого упоминания в тексте. Слово Таблица и ее номер отделяют от названия точкой и пробелом. В конце названия таблицы точку не ставят, если только следующий за названием пояснительный текст не входит в блок полного названия таблицы. Если таблица без номера (единственная в статье), то пишут: Таблица. Название таблицы.

Название таблицы должно быть лаконичным и информативным; как правило, оно должно состоять из одного предложения.

Термины в названии таблицы должны соответствовать терминам в тексте.

Головки столбцов определяют их содержание; каждый столбец таблицы, в том числе и боковик, должен быть снабжен головкой. У каждого столбца должно быть название.

Крайний левый столбец таблицы называется боковиком. Подобно всем остальным столбцам боковик снабжен головкой. Головки строк боковика должны содержать информацию, относящуюся ко всей строке таблицы.

Поле таблицы содержит информацию, которую автор хочет представить читателю. Такой информацией могут быть числа, текст или символы. Каждый элемент информации в таблице расположен на пересечении вертикальной и горизонтальной граф, образующем поле (ячейку, клетку) таблицы. Строки текста в полях располагают не по нижней, а по верхней линии. Если в поле таблицы вводят текстовые данные, необходимо, чтобы они были максимально сжатыми и четкими.

Критерии оценивания	Максим. кол-во баллов
Осмысление прочитанного материала	3
Умение выделять и формулировать главную мысль	5
Лаконичность выражения мысли	5
Наличие библиографических данных источника	3
Общее кол-во баллов	
Оценка	

- 16-13 -оценка «5»;
- 12-9 – оценка «4»;
- 8-7 – оценка «3»;
- 6 и ниже -оценка «2»

Самостоятельная работа 3. Использование разветвляющихся и циклических алгоритмов при решении задач.

Методические указания по выполнению задания:

Начертить блок схему в соответствии с индивидуальным заданием и в соответствии с ГОСТ 10.002-80 ЕСПД, ГОСТ 10.003-80 ЕСПД руководствуясь приведенными в предыдущей работе рекомендациями в MSVisio.

Задание 1. Варианты индивидуальных заданий по теме «Разветвляющиеся алгоритмы»

1. Имеется пронумерованный список деталей: 1) шуруп, 2) гайка, 3) винт, 4) гвоздь, 5) болт. Составить программу, которая по номеру детали выводит на экран ее название.
2. Напечатать 1, если сумма цифр введенного пользователем трехзначного числа – нечетное число, 0 – в противном случае.
3. Даны целые числа m , n . Если числа не равны, то заменить каждое из них одним и тем же числом, равным большему из исходных, а если равны, то заменить числа нулями.
4. Определить, равен ли квадрат заданного трехзначного числа кубу суммы цифр этого числа.
5. Написать программу, которая по номеру месяца выдает название следующего за ним месяца (при $m=1$ получаем февраль, 4 – май).

6. Подсчитать количество отрицательных чисел среди чисел a, b, c .
7. Найти $\min\{\min(a,b), \max(c,d)\}$.
8. Подсчитать количество целых чисел среди чисел a, b, c .
9. Определить, делителем каких чисел a, b , является число k .
10. Подсчитать количество четных чисел среди чисел a, b, c .
11. Перераспределить значения переменных x и y , чтобы в x оказалось большее из этих значений, а в y – меньшее.
12. Подсчитать количество нечетных чисел среди чисел a, b, c .
13. Найти $\min\{\max(a,b), \max(c,d)\}$.
14. Определить, делителем каких чисел a, b , является число k .
15. Определить, равен ли куб заданного четырехзначного числа квадрату суммы цифр этого числа.
16. Найти $\max\{\min(a,b), \min(c,d)\}$.
17. Даны действительные числа a, b, c . Удвоить эти числа, если $a = b$ и $b < c$, и заменить их абсолютными значениями, если это не так.
18. Для целого числа k от 1 до 99 напечатать фразу "Мне k лет", учитывая при этом, что при некоторых значениях k слово "лет" надо заменить на слово "год" или "года". Например, 11 лет, 22 года, 51 год.
19. Написать программу, которая по введенному номеру единицы измерения (1 - дециметр, 2 - километр, 3 - метр, 4 - миллиметр, 5 - сантиметр) или длине отрезка L выдавала бы соответствующее значение длины отрезка в метрах.
20. Определить, равен ли куб заданного трехзначного числа квадрату суммы цифр этого числа.
21. Найти $\min\{\max(a,b), \min(c,d)\}$.
22. Напечатать 1, если сумма цифр введенного пользователем четырехзначного числа – четное число, 0 – в противном случае.
23. Подсчитать количество дробных чисел среди чисел a, b, c .
24. Даны действительные числа a, b, c . Утроить эти числа, если $a > b > c$, и заменить их абсолютными значениями, если это не так.
25. Даны действительные числа a, b, c . Удвоить эти числа, если $a < b < c$, и заменить их абсолютными значениями, если это не так.
26. Если сумма введенных пользователем целых чисел больше нуля, заменить каждое из них произведением этих чисел.
27. Подсчитать количество четных чисел среди чисел a, b, c, d .
28. Даны действительные числа a, b, c . Сложить эти числа, если $a > b$ и $b < c$, и заменить их абсолютным значением разности этих чисел, если это не так.
29. Даны действительные числа a, b, c . Удвоить эти числа, если $a > b$ и $b > c$, и заменить их нулями, если это не так.

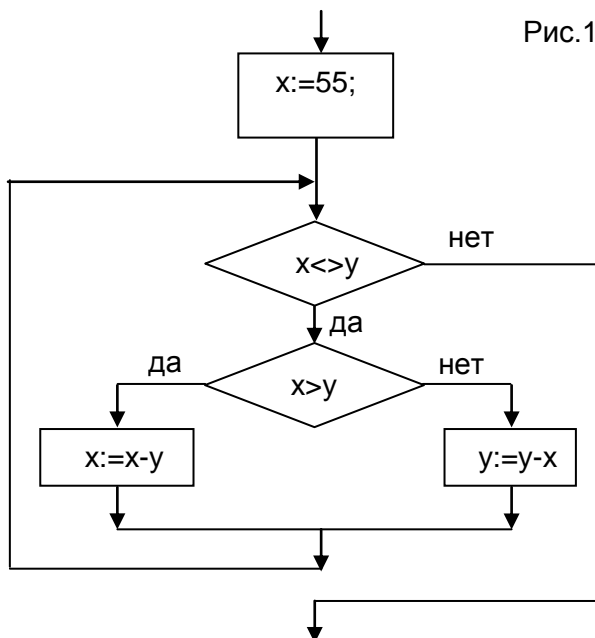


Рис.1.

30. Определить, равен ли квадрат заданного трехзначного числа кубу суммы цифр этого числа.

Задание 2. Задания по теме «Циклические алгоритмы»

Пример выполнения задания: Определите значение целочисленной переменной x после выполнения следующего фрагмента блок-схемы (см. рис.1)

- 1) 1;
- 2) 5;
- 3) 10;
- 4) 15.

Решение. В блок-схеме присутствует повторяющаяся последовательность действий (цикл). Для того, чтобы не ошибиться при выполнении блок-схемы, составим таблицу (см. Таблицу 1), в которую будем заносить значения переменных и результаты проверки условий на каждом шаге.

Таблица 1.

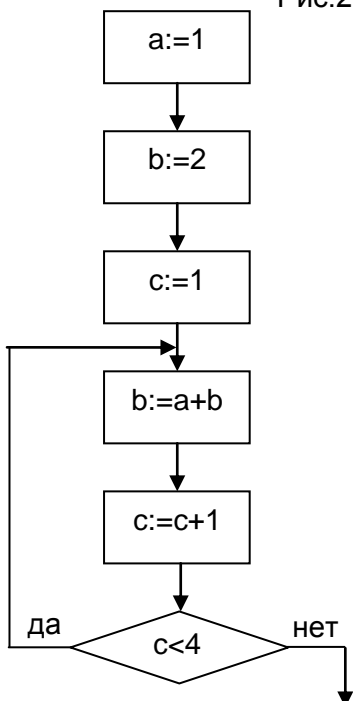
№ итерации	Значение x	Значение y	$x <> y$	$x > y$
0	55	75	$55 <> 75$ – да Выполняем тело цикла	$55 > 75$ – нет, $y := y - x = 75 - 55 = 20$
1	55	20	$55 <> 20$ – да Выполняем тело цикла	$55 > 20$ – да $x := x - y = 55 - 20 = 35$
2	35	20	$35 <> 20$ – да Выполняем тело цикла	$35 > 20$ – да $x := x - y = 35 - 20 = 15$
3	15	20	$15 <> 20$ – да Выполняем тело цикла	$15 > 20$ – нет, $y := y - x = 20 - 15 = 5$
4	15	5	$15 <> 5$ – да Выполняем тело цикла	$15 > 5$ – да $x := x - y = 15 - 5 = 10$
5	10	5	$10 <> 5$ – да Выполняем тело цикла	$10 > 5$ – да $x := x - y = 10 - 5 = 5$
6	5	5	$5 <> 5$ – нет Выход из цикла; завершение алгоритма	

Таким образом, переменная x после выполнения данного фрагмента программы приняла значение 5, что соответствует ответу под номером 2.

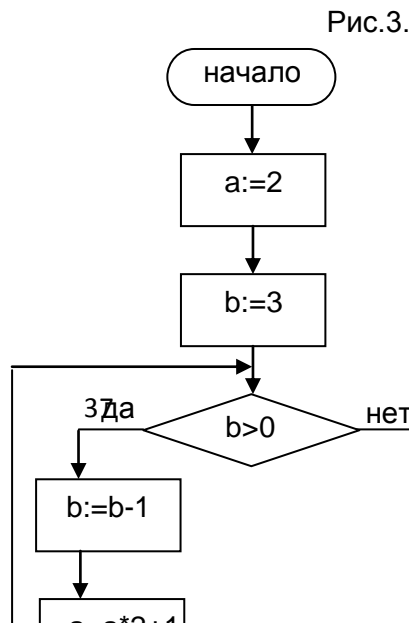
Ответ: 2.

Задачи для самостоятельного решения:

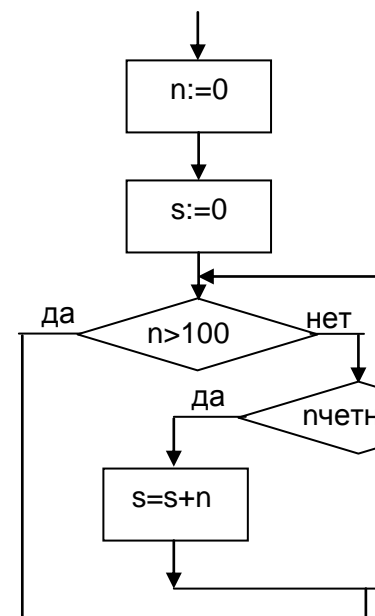
1. Определите значение переменной b после выполнения следующего фрагмента алгоритма (см. рис.2):



2. Определите значение переменной a после выполнения алгоритма (см. рис.3):



3. Определите значение переменной s после выполнения фрагмента алгоритма (см. рис. 4).



4. Контрольно-оценочные материалы для промежуточной аттестации по дисциплине

Оценка освоения дисциплины предусматривает использование накопительной системы оценивания и проведение дифференцированного зачета.

КОМ предназначен для контроля и оценки результатов освоения дисциплины Теория алгоритмов по специальности СПО 09.02.03 Программирование в компьютерных системах базового уровня подготовки

Умения

- У1 разрабатывать алгоритмы для конкретных задач;
- У2 определять сложность работы алгоритмов;

Знания

- З1 основные модели алгоритмов;
- З2 методы построения алгоритмов
- З3 методы вычисления сложности работы алгоритмов

КРИТЕРИИ ОЦЕНКИ

Критерии и шкала оценки тестирования на этапе текущего контроля

На этапе текущего контроля студентам на лабораторной работе, предлагается выполнить тесты по темам дисциплины. Преподаватель определяет студентам исходные данные для подготовки к тестированию, название темы, вопросы, по которым будут задания в тестовой форме, и источники информации для подготовки.

Каждому студенту отводится на тестирование время, соответствующее количеству тестовых заданий. До окончания теста студент может еще раз просмотреть все свои ответы на задания и при необходимости внести коррективы.

При прохождении тестирования пользоваться конспектами лекций, учебниками, и иными материалами не разрешено.

Оценка «5» ставится, если: студент выполняет правильно 86-100 % тестовых заданий.

Оценка «4» ставится, если: студент выполняет правильно 71-85 % тестовых заданий.

Оценка «3» ставится, если: студент выполняет правильно 51-70% тестовых заданий.

Оценка «2» ставится, если: студент выполняет правильно до 50% тестовых заданий

Критерии оценки письменного опроса

Письменные опросы проводятся во время лекций, практических занятий и возможны при проведении промежуточной аттестации в качестве дополнительного испытания при недостаточности результатов тестирования. Основные вопросы для опроса доводятся до сведения студентов на предыдущем занятии.

Развернутый ответ студента должен представлять собой связное, логически последовательное сообщение на заданную тему, показывать его умение применять определения, правила в конкретных случаях.

По результатам ответа **«отлично»** выставляется студенту, если содержание ответа соответствует освещаемому вопросу, полностью раскрыта в ответе тема, ответ структурирован, даны правильные, аргументированные ответы на уточняющие вопросы, демонстрируется высокий уровень участия в дискуссии.

По результатам ответа **«хорошо»** выставляется студенту, если содержание ответа соответствует освещаемому вопросу, полностью раскрыта в ответе тема, даны правильные, аргументированные ответы на уточняющие вопросы, но имеются неточности, при этом ответ неструктурирован и демонстрируется средний уровень участия в дискуссии.

По результатам ответа **«удовлетворительно»** выставляется студенту, если содержание ответа соответствует освещаемому вопросу, но при полном раскрытии темы имеются неточности, даны правильные, но не аргументированные ответы на уточняющие вопросы, демонстрируется низкий уровень участия в дискуссии, ответ неструктурирован, информация трудна для восприятия.

По результатам ответа **«неудовлетворительно»** выставляется студенту, если содержание ответа соответствует освещаемому вопросу, но тема в ответе не полностью раскрыта, демонстрируется слабое владение категориальным аппаратом, происходит подмена понятий, даны неправильные, не аргументированные ответы на уточняющие вопросы, участие в дискуссии полностью отсутствует, ответ неструктурирован, информация трудна для восприятия.

При несоответствии содержания ответа, освещаемому вопросу студент также получает **«неудовлетворительно»**.

Критерии оценки практической работы

Оценка **«отлично»** выставляется, если: выполнены все задания практического занятия, студент четко и без ошибок ответил на все контрольные вопросы.

Оценка **«хорошо»** выставляется, если: выполнены все задания практического занятия, студент ответил на все контрольные вопросы с замечаниями.

Оценка **«удовлетворительно»** выставляется, если: выполнены все задания практического занятия с замечаниями; студент ответил на все контрольные вопросы с замечаниями.

Оценка **«неудовлетворительно»** выставляется, если: студент не выполнил или выполнил неправильно задания практического занятия, студент ответил на контрольные вопросы с ошибками или не ответил на контрольные вопросы.

Критерии оценивания самостоятельной работы

оценка «5» :

- решенная задача полностью соответствует спецификации;
- обучающийся может обосновать принятое решение.

оценка «4»:

- решенная задача в основном соответствует спецификации, отклонения от спецификации допущены при реализации второстепенных подзадач;
- обучающийся может объяснить принятое решение.

оценка «3»:

- решенная задача имеет отклонения от спецификации при решении основной задачи;
- обучающийся может объяснить функционирование программы по ее исходному тексту.

оценка «2»:

- разработка задачи не завершена;
- решенная задача не соответствует спецификации или обучающийся не может объяснить функционирование программного модуля.

Критерии и шкала оценивания промежуточной аттестации (дифференцированный зачет)

Дифференцированный зачет выставляется по совокупному результату текущего контроля успеваемости студента по дисциплине:

«отлично» - выставляется при условии, если студент выполнил все практические занятия с оценкой не ниже «хорошо и отлично», результаты тестирования и устного опроса в совокупности также оценены с оценкой не ниже «хорошо и отлично» при этом совокупный средний балл оценок не ниже 4,6.

«хорошо» - выставляется при условии, если студент выполнил все практические занятия с оценкой не ниже «хорошо, отлично, удовлетворительно», результаты тестирования и устного опроса в совокупности также оценены с оценкой не ниже «хорошо, отлично, удовлетворительно», при этом совокупный средний балл оценок не ниже 3,8.

«удовлетворительно» - выставляется при условии, если студент выполнил все практические занятия с оценкой не ниже «удовлетворительно», результаты тестирования и устного опроса в совокупности также оценены с оценкой не ниже «удовлетворительно», при этом совокупный средний балл оценок ниже 3,8.

«неудовлетворительно» - не выполнены условия для получения оценки «удовлетворительно».