

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра теоретической и прикладной информатики
Кафедра прикладной математики

“УТВЕРЖДАЮ”
ДЕКАН ФПМИ
д.т.н. Тимофеев В. С.
“ ____ ” _____ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Теория языков программирования и методы трансляции

Образовательная программа: 02.03.03 Математическое обеспечение и администрирование информационных систем, профиль: Математическое и программное обеспечение информационных технологий

Курс: 3, семестр : 6

Факультет прикладной математики и информатики,

		Семестр
№	Вид деятельности	6
1	Всего зачетных единиц (кредитов)	3
2	Всего часов	108
3	Всего занятий в контактной форме, час.	61
4	Лекции, час.	36
5	Практические занятия, час.	0
6	Лабораторные занятия, час.	18
7	из них в активной и интерактивной форме, час.	14
8	Аттестация, час.	2
9	Консультации, час.	5
10	Самостоятельная работа, час.	47
11	Виды самостоятельной работы (курсовой проект, курсовая работа, РГЗ, подготовка к контрольной работе)	РГЗ
12	Вид аттестации	ДЗ

Рабочая программа составлена на основании федерального государственного образовательного стандарта высшего образования по направлению (специальности): 02.03.03 Математическое обеспечение и администрирование информационных систем

ФГОС введен в действие приказом №222 от 12.03.2015 г. , дата утверждения: 07.04.2015 г.

Место дисциплины в структуре учебного плана: Блок 1, вариативная, по выбору студента

Рабочая программа разработана на основе компетентностной модели выпускника по направлению (специальности): 02.03.03 Математическое обеспечение и администрирование информационных систем

Рабочая программа обсуждена на заседании кафедры

ТПИ, протокол заседания кафедры №4 от 20.06.2017

ПМт, протокол заседания кафедры №6 от 20.06.2017

Утверждена на совете факультета прикладной математики и информатики, протокол № 6 от 21.06.2017

Программу разработал:

доцент, к.т.н. Еланцева И. Л.

Заведующий кафедрой:

доцент, д.т.н. Чубич В. М.

профессор, д.т.н. Соловейчик Ю. Г.

Ответственный за образовательную программу:

заведующий кафедрой Чубич В. М.

1. Внешние требования

Таблица 1.1

Компетенция ФГОС: ОПК.2 способность применять в профессиональной деятельности знания математических основ информатики; в части следующих результатов обучения:	
33. Знать методы лексического, синтаксического анализа, генерации кода	
34. Знать теорию формальных языков и грамматик	
у2. Уметь разрабатывать и реализовывать все этапы трансляции	
Компетенция НГТУ: ПК.11.В/П способность формировать суждения о проблемах современной информатики; в части следующих результатов обучения:	
у1. Уметь разрабатывать грамматику и распознаватель для транслятора с языка высокого уровня	

2. Требования НГТУ к результатам освоения дисциплины

Таблица 2.1

Результаты изучения дисциплины по уровням освоения (иметь представление, знать, уметь, владеть)	Формы организации занятий
<i>Теория языков программирования и методы трансляции</i>	
ОПК.2.33 Знать методы лексического, синтаксического анализа, генерации кода	
1. Знать методы логического, синтаксического анализа, генерации кода	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.2.34 Знать теорию формальных языков и грамматик	
2. Знать теорию формальных языков и грамматик	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.2.у2 Уметь разрабатывать и реализовывать все этапы трансляции	
3. Уметь разрабатывать и реализовывать все этапы трансляции	Лекции; Лабораторные работы; Самостоятельная работа
ПК.11.В/П.у1 Уметь разрабатывать грамматику и распознаватель для транслятора с языка высокого уровня	
4. Уметь разрабатывать грамматику и распознаватель для транслятора с языка высокого уровня	Лекции; Лабораторные работы; Самостоятельная работа
5. Знать особенности различных языков программирования	Лекции; Самостоятельная работа

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 6				
Дидактическая единица: методы и основные этапы трансляции				
1. Структуры трансляторов и методы их реализации. Цели и задачи трансляции. Основные этапы трансляции	0	2	1	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции

2. Лексический анализатор. Понятие лексического анализатора. Задача преобразования недетерминированного конечного автомата без памяти в лексический анализатор. Алгоритм преобразования недетерминированного конечного автомата в детерминированный. Методика построения лексического анализатора для нескольких групп слов. Обработка ошибок при лексическом анализе. Понятие лексемы. Формирование лексем и методы организации информационных таблиц транслятора. Структура и основные алгоритмы лексического анализатора	0	2	1, 2	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции
6. Генерация кода. Представление промежуточной программы. Понятие постфиксной формы записи, её связь с деревом грамматического разбора, задача преобразования последовательности лексем в постфиксную запись. Модели процесса генерации кода. Системы построения трансляторов.	0	2	1, 3	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции
Дидактическая единица: синтаксис, семантика, формальные способы описания языков программирования				
3. Синтаксический анализ. Понятие дерева грамматического разбора. Восстановление дерева разбора, как основная задача синтаксического анализа. Нисходящие и восходящие группы методов синтаксического анализа, их основные идеи. Понятие конечного автомата со стековой памятью в качестве основы синтаксического анализатора	0	1	1, 2, 4	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции

4. Основная идея алгоритма нисходящего синтаксического анализа. Критерии выбора грамматики для реализации этого алгоритма. Понятие S-грамматики для данного языка. Понятие и определение LL(1)-грамматики. Алгоритм определения принадлежности грамматики к классу LL(1). Задача преобразования LL(1)-грамматики в управляющую таблицу нисходящего синтаксического анализатора. Обработка ошибок при нисходящем синтаксическом анализе, методы нейтрализации ошибок.	1	6	1, 4	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции участвовать в дискуссии о преимуществах и недостатках нисходящего разбора
5. Основные идеи восходящих методов синтаксического анализа. Операции сдвига и свёртки, состояние стека автомата до и после их выполнения. Задача определения множества состояний восходящего анализатора. Понятие конфигурации, связь между подмножествами конфигураций и состоянием автомата. Алгоритм построения подмножеств эквивалентных конфигураций. Разметка грамматики. Структура управляющей таблицы восходящего анализатора. Обработка ошибок при восходящем синтаксическом анализе, методы нейтрализации ошибок	1	5	1, 4	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции участвовать в дискуссии о преимуществах и недостатках восходящего разбора
Дидактическая единица: Основные понятия языков программирования				
7. Основные понятия языков программирования. Основные понятия языков программирования, синтаксис, семантика. История языков программирования. Критерии сравнения языков программирования. Стандартизация языков. Среда программирования. Влияние машинной архитектуры и среды программирования на развитие языка. Формальные способы описания языков программирования	1	6	5	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции участвовать в дискуссии по теме лекции
Дидактическая единица: типы данных, способы и механизмы управления данными				

8. Типы данных. Элементарные типы данных в языках программирования. Контроль и преобразование типов. Скалярные типы данных (численные, булевы, перечисления) и их реализация в различных языках (на примерах Фортран, C++, PL/1, Java). Составные типы данных (строки, указатели, файлы).	1	4	5	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции участвовать в дискуссии по теме лекции
9. Структурированные типы данных. Структурированные типы данных. Векторы и массивы. Сечения массивов и их реализация с использованием дескрипторов в языке Фортран. Записи, списки. Выполняемые объекты данных. Абстрактные типы данных. Инкапсуляция при помощи подпрограмм в языках модульного программирования. Определение типов. Параметризованные типы (на примере языка Паскаль). Наследование.	0	4	5	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции
Дидактическая единица: конструкции распределенного и параллельного программирования				
10. Управление последовательностью выполнения программы. Явное и неявное управление последовательностью действий. Последовательность выполнения в объектно-ориентированном языке (на примере C++). Рекурсия. Виртуальные функции. Управление распределением памяти. Конструкции распределенного и параллельного программирования	1	4	5	конспектировать объяснения преподавателя или воспользоваться изданным конспектом лекций, участвовать в обсуждении примеров, задавать вопросы по теме лекции участвовать в дискуссии по теме лекции

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 6				
Дидактическая единица: методы и основные этапы трансляции				
1. Проектирование и реализация таблиц, используемых в трансляторе	2	4	3	сформулировать цели и задачи проекта; определить вид, объем и источник исходных данных; выбрать структуру таблиц; реализовать таблицы; оформить отчет и защитить работу.

2. Разработка и реализация блока лексического анализа (сканер)	2	4	1, 2, 3, 4	сформулировать цели и задачи проекта; определить вид, структуру входных и выходных данных; разработать и реализовать сканер на языке высокого уровня; оформить отчет и защитить работу
4. Разработка и реализация блока генерации кода	3	6	1, 3	сформулировать цели и задачи проекта; определить вид, структуру входных и выходных данных; разработать и реализовать генератор на языке высокого уровня; оформить отчет и защитить работу
Дидактическая единица: синтаксис, семантика, формальные способы описания языков программирования				
3. Разработка и реализация блока синтаксического анализа	2	4	3, 4	сформулировать цели и задачи проекта; определить вид, структуру входных и выходных данных; классифицировать входной язык, определить необходимость преобразования языка к требуемому виду, при необходимости выполнить эти преобразования; выбрать стратегию анализа разработать и реализовать анализатор на языке высокого уровня; оформить отчет и защитить работу

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 6				
1	РГЗ	1, 3, 5	20	2
По теме, предложенной преподавателем, студент самостоятельно находит учебную литературу и изучает некоторый аспект процесса трансляции, разрабатывает программу, реализующую изученные методы трансляции, оформляет отчет.: Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011 Еланцева И. Л. Языки программирования и методы трансляции [Электронный ресурс] : электронный учебно-методический комплекс / И. Л. Еланцева, М. Э. Рояк, Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2013]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000183565 . - Загл. с экрана.				
2	Подготовка к занятиям	1, 2, 3, 4, 5	20	2

Подготовка к лекционным занятиям состоит в повторении уже прочитанных лекций. Подготовка к лабораторным занятиям включает повторении уже прочитанных лекций, разработке формальных грамматик, распознавателей, написании программы, реализующей процессы трансляции, подготовку отчета., подробная информация приведена в приложении №1 : Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011				
3	Подготовка к аттестации	1, 2, 5	7	1
Подготовка к аттестации включает повторение теоретического материала по конспекту лекций и рекомендуемой литературе., подробная информация приведена в приложении №1 : Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011				

5. Технология обучения

Для организации и контроля самостоятельной работы обучающихся, а также проведения консультаций применяются информационно-коммуникационные технологии (табл. 5.1).

Таблица 5.1

Деятельность	Информационно-коммуникационные технологии
Информирование	e-mail
Консультирование	e-mail
Контроль	e-mail
Размещение учебных материалов	ЭБС

Таблица 5.2

Активные и интерактивные формы проведения занятий

№	Наименование активных форм	Коды формируемых компетенций
1	Дискуссия	ОПК.2; ПК.11.В/П
Формируемые умения: з3. Знать методы лексического, синтаксического анализа, генерации кода; з4. Знать теорию формальных языков и грамматик ; у1. Уметь разрабатывать грамматику и распознаватель для транслятора с языка высокого уровня; у2. Уметь разрабатывать и реализовывать все этапы трансляции		
Краткое описание применения: сравнительный анализ существующих методов трансляции		
Подробная информация об использовании технологии приводится в "Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011"		

6. Правила аттестации обучающихся по учебной дисциплине

Для аттестации обучающихся по дисциплине используется балльно-рейтинговая система (БРС), позволяющая выставять оценки по традиционной шкале и 15-уровневой ECTS. Краткая информация о БРС приведена в табл. 6.1.

Таблица 6.1

Оцениваемые виды деятельности обучающихся	Мин. балл	Максимальный балл
Семестр: 6		
Лабораторная №2: 1	8	16
Контролирующие материалы приводятся в "Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011"		
Лабораторная №3: 2	8	16

Контролирующие материалы приводятся в "Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011"		
Лабораторная №4: 3	8	16
Контролирующие материалы приводятся в "Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011"		
Лабораторная №5: 4	8	16
Контролирующие материалы приводятся в "Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011"		
РГЗ:	8	16
Зачет:	10	20

В таблице 6.2 представлено соответствие форм контроля заявляемым требованиям к результатам освоения дисциплины.

Таблица 6.2

Коды компетенций ФГОС	Результаты обучения	Формы контроля		
		Защита ЛР	Защита РГЗ	Зачет
ОПК.2	33. Знать методы лексического, синтаксического анализа, генерации кода	+	+	
	34. Знать теорию формальных языков и грамматик	+	+	
	у2. Уметь разрабатывать и реализовывать все этапы трансляции	+	+	
	ПК.11.В/П у1. Уметь разрабатывать грамматику и распознаватель для транслятора с языка высокого уровня	+	+	+

Фонд оценочных средств по дисциплине представлен в приложении № 1 к рабочей программе.

7. Литература

Основная литература

1. Свердлов С. З. Языки программирования и методы трансляции : учебное пособие для вузов / С. З. Свердлов. - СПб., 2007. - 637 с. + 1 CD-ROM. - Издательская программа 300 лучших учебников для высшей школы.
2. Карпов Ю. Г. Теория и технология программирования. Основы построения трансляторов : учебное пособие для вузов по направлениям подготовки бакалавров и магистров 553000 - "Системный анализ и управление" и 552800 - "Информатика и вычислительная техника" / Ю. Г. Карпов. - СПб., 2005. - 270 с. : ил., табл.
3. Еланцева И. Л. Языки программирования и методы трансляции. Раздел "Методы трансляции" : [конспект лекций] / И. Л. Еланцева, И. А. Полетаева ; Новосиб. гос. техн. ун-т. - Новосибирск, 2008. - 97, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000077650

Дополнительная литература

1. Хантер Р. Проектирование и конструирование компиляторов / Р. Хантер; пер. с англ. С. М. Круговой под ред. В. М. Савинкова. - М., 1984. - 231, [1] с. : ил.
2. Ахо А. В. Компиляторы : Принципы, технологии, инструменты / А. Ахо, Р. Сети, Д. Ульман. - М., 2001. - 766 с.

Интернет-ресурсы

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>

4. ЭБС "Znanium.com" : <http://znanium.com/>

5. :

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

1. Языки программирования и методы трансляции : методические указания / Новосиб. гос. техн. ун-т ; [сост.: И. Л. Еланцева, И. А. Полетаева]. - Новосибирск, 2011

2. Еланцева И. Л. Языки программирования и методы трансляции [Электронный ресурс] : электронный учебно-методический комплекс / И. Л. Еланцева, М. Э. Рояк, Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2013]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000183565. - Загл. с экрана.

8.2 Специализированное программное обеспечение

1 Microsoft Visual C++

2 Microsoft Office

9. Материально-техническое обеспечение

Презентационное оборудование

№	Наименование	Назначение
1	Презентационное оборудование (мультимедиа-проектор, экран, компьютер для управления)	Проведение лекционных
2	Компьютерный класс (Компьютеры объединены в локальную сеть с выходом в Internet)	выполнение лабораторных работ и РГР

Правила аттестации студентов по учебной дисциплине Теория языков программирования и методы трансляции

Текущая аттестация студента проводится по результатам выполнения им лабораторных работ и расчетно-графического задания.

Баллы за каждую из четырех лабораторных работ выставляются по правилам, приведенным в таблице 1.

Таблица 1.

Вид деятельности	Минимальное количество баллов	Максимальное количество баллов
Выполнение лабораторной работы (оценивается полнота выполнения задания, работоспособность разработанного программного обеспечения, своевременность* выполнения лабораторной работы)	3	6
Оформление отчета (оценивается полнота набора тестов, полнота отчета, его соответствие требованиям оформления, своевременность* выполнения отчета)	2	4
Защита (оценивается полнота и глубина теоретических знаний, умение применить знания на практике, своевременность* защиты)	3	6
Всего за лабораторную работу	8	16

* график выполнения и защиты лабораторных работ приведен ниже

Минимальное допустимое количество баллов за лабораторную работу получает студент, выполнивший задание не полно, оформивший отчет с недостатками и показавший на защите некоторое владение теоретическими и практическими навыками. Студент, не выполнивший и не защитивший лабораторную работу, не допускается к выполнению следующей лабораторной работы.

График выполнения и защиты лабораторных работ:

- 1 – 4 неделя – 1-я лабораторная работа;
- 5 – 8 неделя – 2-я лабораторная работа;
- 9 – 14 неделя – 3-я лабораторная работа.
- 15 – 18 неделя – 4-я лабораторная работа.

Оценка за контрольные недели выставляется в соответствии с таблицей 2.

Таблица 2.

Номер недели	7 контрольная неделя	12 контрольная неделя
--------------	----------------------	-----------------------

Оценка за контрольную неделю	0	1	2	0	1	2
Текущий рейтинг студента	Менее 8	От 8 до 12	14 и более	Менее 22	От 22 до 32	28 и более
Количество зачтенных лабораторных заданий	0	1	2	1	2	3*

* лабораторная работа выполнена, но не защищена

Баллы за каждую расчетно-графическую работу выставляются по правилам, приведенным в таблице 3.

Таблица 3.

Вид деятельности	Минимальное количество баллов	Максимальное количество баллов
Реферат (оценивается полнота раскрытия темы и современного состояния проблемы) и защита (оценивается полнота и глубина теоретических знаний, умение применить знания на практике)	4	8
Практическая часть (оценивается качество разработанного программного обеспечения, полнота тестов)	4	8
Всего за расчетно-графическую работу	8	16

Зачет в шестом семестре сдается в виде письменных ответов тест. К зачету допускаются студенты, выполнившие и защитившие все 4 лабораторные работы РГР и набравшие не менее 40 баллов. Максимальное количество баллов за теоретический зачет – 20 баллов, минимальное – 10. Оценивается количество полных и правильных ответов на задания теста. Минимальное количество баллов за теоретический зачет получает студент, ответивший правильно на 10 заданий теста.

Соответствие разных шкал оценивания приведено в таблице 4

Таблица 4.

Характеристика работы студента	Диапазон баллов рейтинга	Оценка ECTS	Традиционная (4-уровневая) шкала оценки	
«Отлично» – работа высокого качества, уровень выполнения отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые	98-100	A+	отлично	зачтено
	93-97	A		

практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	90-92	A-		
«Очень хорошо» – работа хорошая, уровень выполнения отвечает большинству требований, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному	87-89	B+		
	83-86	B		
	80-82	B-	хорошо	
«Хорошо» – уровень выполнения работы отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки	77-79	C+		
	73-76	C		
70-72	C-	удовлетворительно		
«Удовлетворительно» – уровень выполнения работы отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном	67-69	D+	удовлетворительно	зачтено
	63-66	D		
	60-62	D-		

сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками				
«Посредственно» – работа слабая, уровень выполнения не отвечает большинству требований, теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному	50-59	E		
«Неудовлетворительно» (с возможностью пересдачи) – теоретическое содержание курса освоено частично, необходимые практические навыки работы не сформированы, большинство предусмотренных программой обучения учебных заданий не выполнено, либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	25-49	FX	неудовлетворительно	незачтено
«Неудовлетворительно» (без возможности пересдачи) – теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, все выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к какому-либо значимому повышению качества выполнения учебных заданий	0-24	F		

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра прикладной математики
Кафедра теоретической и прикладной информатики

[illegible]

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Теория языков программирования и методы трансляции

Образовательная программа: 02.03.03 Математическое обеспечение и администрирование информационных систем, профиль: Математическое и программное обеспечение информационных технологий

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Теория языков программирования и методы трансляции приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.2 способность применять в профессиональной деятельности знания математических основ информатики	з3. Знать методы лексического, синтаксического анализа, генерации кода	Генерация кода. Представление промежуточной программы. Понятие постфиксной формы записи, её связь с деревом грамматического разбора, задача преобразования последовательности лексем в постфиксную запись. Модели процесса генерации кода. Системы построения трансляторов. Лексический анализатор. Понятие лексического анализатора. Задача преобразования недетерминированного конечного автомата без памяти в лексический анализатор. Алгоритм преобразования недетерминированного конечного автомата в детерминированный. Методика построения лексического анализатора для нескольких групп слов. Обработка ошибок при лексическом анализе. Понятие лексем. Формирование лексем и методы организации информационных таблиц транслятора. Структура и основные алгоритмы лексического анализатора Основная идея алгоритма нисходящего синтаксического анализа. Критерии выбора грамматики для реализации этого алгоритма. Понятие S-грамматики для данного языка. Понятие и определение LL(1)-грамматики. Алгоритм определения принадлежности грамматики к классу LL(1). Задача преобразования LL(1)-грамматики в управляющую таблицу нисходящего синтаксического анализатора. Обработка ошибок при нисходящем синтаксическом анализе, методы нейтрализации ошибок. Основные идеи восходящих методов синтаксического	Отчет по лабораторной работе №2, Контрольные вопросы 1-7 Отчет по лабораторной работе №3, Контрольные вопросы 1-8, Отчет по лабораторной работе №4, Контрольные вопросы 1-7 РГЗ (темы 1, 6-12, 17-21)	Зачет, задания теста 13-20

		анализа. Операции сдвига и свёртки, состояние стека автомата до и после их выполнения. Задача определения множества состояний восходящего анализатора. Понятие конфигурации, связь между подмножествами конфигураций и состоянием автомата. Алгоритм построения подмножеств эквивалентных конфигураций. Разметка грамматики. Структура управляющей таблицы восходящего анализатора. Обработка ошибок при восходящем синтаксическом анализе, методы нейтрализации ошибок. Разработка и реализация блока генерации кода. Разработка и реализация блока лексического анализа (сканер). Синтаксический анализ. Понятие дерева грамматического разбора. Восстановление дерева разбора, как основная задача синтаксического анализа. Нисходящие и восходящие группы методов синтаксического анализа, их основные идеи. Понятие конечного автомата со стековой памятью в качестве основы синтаксического анализатора. Структуры трансляторов и методы их реализации. Цели и задачи трансляции. Основные этапы трансляции		
ОПК.2	34. Знать теорию формальных языков и грамматик	Лексический анализатор. Понятие лексического анализатора. Задача преобразования недетерминированного конечного автомата без памяти в лексический анализатор. Алгоритм преобразования недетерминированного конечного автомата в детерминированный. Методика построения лексического анализатора для нескольких групп слов. Обработка ошибок при лексическом анализе. Понятие лексем. Формирование лексем и методы организации информационных таблиц транслятора. Структура и основные алгоритмы лексического анализатора. Разработка и реализация блока лексического анализа (сканер). Синтаксический анализ.	Отчет по лабораторной работе №1, Контрольные вопросы 1-2, Отчет по лабораторной работе №2, Контрольные вопросы 1-7 РГЗ (темы 5, 13-16)	Зачет, задания теста 1-12

		Понятие дерева грамматического разбора. Восстановление дерева разбора, как основная задача синтаксического анализа. Нисходящие и восходящие группы методов синтаксического анализа, их основные идеи. Понятие конечного автомата со стековой памятью в качестве основы синтаксического анализатора		
ОПК.2	у2. Уметь разрабатывать и реализовывать все этапы трансляции	Генерация кода. Представление промежуточной программы. Понятие постфиксной формы записи, её связь с деревом грамматического разбора, задача преобразования последовательности лексем в постфиксную запись. Модели процесса генерации кода. Системы построения трансляторов. Проектирование и реализация таблиц, используемых в трансляторе Разработка и реализация блока генерации кода Разработка и реализация блока лексического анализа (сканер) Разработка и реализация блока синтаксического анализа	Отчет по лабораторной работе №3, Контрольные вопросы 1-8, Отчет по лабораторной работе №4, Контрольные вопросы 1-7 РГЗ (темы 2-4, 22-23)	Зачет, задания теста 13-20
ПК.11.В/П способность формировать суждения о проблемах современной информатики	у1. Уметь разрабатывать грамматику и распознаватель для транслятора с языка высокого уровня	Основная идея алгоритма нисходящего синтаксического анализа. Критерии выбора грамматики для реализации этого алгоритма. Понятие S-грамматики для данного языка. Понятие и определение LL(1)-грамматики. Алгоритм определения принадлежности грамматики к классу LL(1). Задача преобразования LL(1)-грамматики в управляющую таблицу нисходящего синтаксического анализатора. Обработка ошибок при нисходящем синтаксическом анализе, методы нейтрализации ошибок. Основные идеи восходящих методов синтаксического анализа. Операции сдвига и свёртки, состояние стека автомата до и после их выполнения. Задача определения множества состояний восходящего анализатора. Понятие конфигурации, связь между подмножествами конфигураций и состоянием автомата. Алгоритм построения подмножеств эквивалентных конфигураций. Разметка грамматики.	Отчет по лабораторной работе №3, Контрольные вопросы 1-8, Отчет по лабораторной работе №4, Контрольные вопросы 1-7 РГЗ (темы 2-4, 22-23)	Зачет, задания теста 13-20

		Структура управляющей таблицы восходящего анализатора. Обработка ошибок при восходящем синтаксическом анализе, методы нейтрализации ошибок Основные понятия языков программирования. Основные понятия языков программирования, синтаксис, семантика. История языков программирования. Критерии сравнения языков программирования. Стандартизация языков. Среда программирования. Влияние машинной архитектуры и среды программирования на развитие языка. Формальные способы описания языков программирования Разработка и реализация блока лексического анализа (сканер) Разработка и реализация блока синтаксического анализа Синтаксический анализ. Понятие дерева грамматического разбора. Восстановление дерева разбора, как основная задача синтаксического анализа. Нисходящие и восходящие группы методов синтаксического анализа, их основные идеи. Понятие конечного автомата со стековой памятью в качестве основы синтаксического анализатора Структурированные типы данных. Структурированные типы данных. Векторы и массивы. Сечения массивов и их реализация с использованием дескрипторов в языке Фортран. Записи, списки. Выполняемые объекты данных. Абстрактные типы данных. Инкапсуляция при помощи подпрограмм в языках модульного программирования. Определение типов. Параметризованные типы (на примере языка Паскаль). Наследование. Типы данных. Элементарные типы данных в языках программирования. Контроль и преобразование типов. Скалярные типы данных (численные, булевы, перечисления) и их реализация в различных языках (на примерах Фортран, C++, PL/1, Java). Составные типы данных (строки, указатели, файлы). Управление		
--	--	---	--	--

		последовательностью выполнения программы. Явное и неявное управление последовательностью действий. Последовательность выполнения в объектно-ориентированном языке (на примере C++). Рекурсия. Виртуальные функции. Управление распределением памяти. Конструкции распределенного и параллельного программирования		
--	--	---	--	--

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 6 семестре - в форме дифференцированного зачета, который направлен на оценку сформированности компетенций ОПК.2, ПК.11.В/П.

На подготовку к зачету отводится 8 часов.

Зачет проводится в письменной форме, по тестам.

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 6 семестре обязательным этапом текущей аттестации является расчетно-графическое задание РГЗ. Требования к выполнению РГЗ, состав и правила оценки сформулированы в паспорте РГЗ.

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.2, ПК.11.В/П, за которые отвечает дисциплина, на разных уровнях.

Общая характеристика уровней освоения компетенций.

Ниже порогового. Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

Пороговый. Уровень выполнения работ отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

Базовый. Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

Продвинутый. Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Паспорт зачета

по дисциплине «Языки программирования и методы трансляции», 6 семестр

1. Методика оценки

Зачет проводится в письменной форме, по тестам.

Пример теста для зачета

1. **Выберите неправильный ответ.**

К единицам языка относится

- а) Идентификатор;
- б) Зарезервированное слово;
- в) Токен;
- г) Разделитель.

2. **Вставьте два пропущенных слова.**

Лексема – это _____ и ее атрибуты.

3. **Выберите правильный ответ.**

Токен – это

- а) единица языка;
- б) тип лексемы и ее адрес в соответствующей таблице;
- в) элемент транслируемой программы.

4. **Выберите правильный ответ.**

Атрибутами обладают

- а) Токены;
- б) Идентификаторы и константы;
- в) Знаки операций;
- г) Постоянные таблицы.

5. **Классифицируйте таблицы, используемые в трансляторах, заполнив форму**

Постоянные таблицы транслятора	Переменные таблицы транслятора

- а) Таблица констант;
- б) Таблица разделителей;
- в) Таблица идентификаторов;
- г) Таблица знаков операций;
- д) Таблица зарезервированных слов.

6. **Установите соответствие между видом таблицы (левый столбец) и типами операций над ее элементами (правый столбец)**

- | | |
|-----------------------|-----------------------|
| 1) Постоянные таблицы | а) Поиск и исключение |
| 2) Переменные таблицы | б) Поиск и добавление |
| | в) Поиск |
| | г) Добавление |

7. Установите правильную последовательность этапов трансляции

- а) Генерация кода.
- б) Синтаксический анализ.
- в) Лексический анализ.
- г) Оптимизация кода.
- д) Генерация промежуточного кода.

8. Выберите правильный ответ.

Грамматика

- а) формирует токен;
- б) задает правила формирования цепочек языка;
- в) задает правила распознавания принадлежности цепочки данному языку;
- г) задает правила работы сканера.

9. Выберите правильный ответ.

Сканер строится на основе

- а) детерминированного конечного автомата;
- б) недетерминированного конечного автомата;
- в) автомата с магазинной памятью.

10. Выберите все неправильные ответы.

К результатам работы сканера относятся

- а) файл сообщений об обнаруженных лексических ошибках;
- б) файл сообщений об обнаруженных синтаксических ошибках;
- в) файл токенов;
- г) диаграмма состояний автомата.

11. Классифицируйте элементы, составляющие автоматную грамматику и конечный автомат, заполнив форму

Автоматная грамматика	Конечный автомат

- а) Множество терминальных символов;
- б) Начальное состояние;
- в) Множество входного алфавита;
- г) Множество конечных состояний;
- д) Множество правил вывода;
- е) Множество нетерминальных символов.

12. Выберите все верные варианты ответов: «да» или «нет».

Сканер обнаруживает следующие типы ошибок

- а) неверная конструкция языка программирования да/нет
- б) недопустимый символ да/нет
- в) ошибка в последовательности литер, определяющих константу да/нет
- г) деление на нуль да/нет
- д) ошибка при обращении к оперативной памяти да/нет

13. Выберите из представленных грамматик LL(1)-грамматику

$G[A] = \{N = \{A, B, C\}, T = \{b, c, d, z\}\}$

a) $A \rightarrow Bd \mid dC$ $A \rightarrow z$ $B \rightarrow \varepsilon \mid bc$ $C \rightarrow z$	b) $A \rightarrow zA \mid \varepsilon \mid BC$ $B \rightarrow \varepsilon \mid bB$ $C \rightarrow cd \mid \varepsilon$	c) $A \rightarrow zA \mid BC$ $B \rightarrow \varepsilon \mid bB$ $C \rightarrow cC \mid d$
--	--	---

14. Ответьте, используя код:

A	B	C	D
а, б, в	б, в, г	а, в, д	в, г, д

При LL(1)-разборе используются

- а) стек символов;
- б) стек состояний;
- в) файл токенов;
- г) таблица разбора;
- д) очередь состояний.

15. Вставьте пропущенное слово

В таблице LL(1)-разбора в столбце **accept** значение *true* ставится только для состояний, соответствующих _____ символам.

16. Выберите все правильные ответы

Необходимым условием успешного окончания LL(1)-разбора цепочки является условие

- а) дерево разбора двоичное.
- б) входным символом является начальный символ грамматики.
- в) файл токенов прочитан до конца.
- г) стек состояний пуст.
- д) все правила грамматики использованы.

17. Выберите правильный ответ

При LR(1)-разборе в стеке символов сохраняются

- а) только терминальные символы.
- б) только нетерминальные символы.
- в) терминальные и нетерминальные символы.

18. Выберите все правильные ответы.

Генератор кода создает команды выделения памяти для

- а) элементов таблицы идентификаторов.
- б) элементов таблицы ключевых слов.
- в) элементов таблицы LR(1)-разбора.
- г) элементов таблицы LL(1)-разбора.

19. Выберите правильный ответ.

Постфиксной записью, соответствующей выражению $x = (y + z) * (u + w) / v * t$ является

- а) $x \ y \ z \ u + w * + v \ t / * =$
- б) $x \ y \ z + u \ w + * v / t * =$
- в) $x \ y \ z + u * w + v / * t =$

20. Укажите все правильные ответы, используя код:

A	B	C	D
а, б, в	б, в, г	а, в, д	б, г, д

В состав «четверки» входят следующие элементы

- а) операнды
- б) приоритет операции
- в) операция
- г) количество операндов
- д) результат

2. Критерии оценки

За полный и правильный ответ за каждое задание студент получает один балл.

- Зачет считается сдан на **пороговом** уровне, если студент дал полный и правильный ответ не менее чем на 10 вопросов, оценка составляет от 10 до 14 баллов
- Зачет считается сдан на **базовом** уровне, если студент дал полный и правильный ответ не менее чем на 15 вопросов, оценка составляет от 15 до 18 баллов
- Зачет считается сдан на **продвинутом** уровне, если студент дал полный и правильный ответ не менее чем на 19 вопросов, оценка составляет от 19 до 20 баллов

3. Шкала оценки

Зачет считается сданным, если сумма баллов по всем заданиям составляет не менее 10 баллов (из 20 возможных).

Паспорт расчетно-графического задания

по дисциплине «Теория языков программирования и методы трансляции», 6 семестр

1. Методика оценки

В рамках расчетно-графического задания по дисциплине студенты должны самостоятельно изучить этап или метод трансляции в соответствии с темой РГЗ.

При выполнении расчетно-графического задания студенты должны:

- используя конспект лекций и рекомендуемую научную литературу, изучить материал по заданной теме;
- написать реферат;
- программно реализовать исследованные методы, включив их в спроектированный транслятор;
- защитить работу.

Обязательные структурные части РГЗ: Реферат о современном состоянии вопроса, описание реализованных алгоритмов, текст программы на языке C++, тестовые примеры.

2. Критерии оценки

- Работа считается **не выполненной**, если реферат не соответствует современному состоянию вопроса, студент не ориентируется в содержании реферата, оценка составляет менее 8 баллов.
- Работа считается выполненной **на пороговом** уровне, если представлен только реферат, без программной реализации алгоритмов, студент хорошо разбирается в содержании реферата, оценка составляет 8-9 баллов.
- Работа считается выполненной **на базовом** уровне, если представлен содержательный реферат, имеется описание реализованных программно методов и проведенных тестов, студент хорошо разбирается в содержании реферата, оценка составляет 10-14 баллов.
- Работа считается выполненной **на продвинутом** уровне, если РГЗ выполнена и защищена в срок, представлен содержательный реферат, имеется описание и обоснование реализованных программно методов и проведенных тестов, студент свободно разбирается в теме РГЗ, оценка составляет 15-16 баллов.

3. Шкала оценки

В общей оценке по дисциплине баллы за РГЗ(Р) учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Перечень тем РГЗ

1. Прямые методы трансляции. Особенности реализации конструкций языков программирования с использованием прямых методов.
2. Методы оптимизации кода. Оптимизация вычислений с константами.
3. Методы оптимизации кода. Оптимизация выражений.
4. Методы оптимизации кода. Оптимизация циклов.
5. Методы диагностики и исправления ошибок. Лексические ошибки.
6. Методы диагностики и исправления ошибок. Синтаксические ошибки.

7. Методы диагностики и исправления ошибок. Ошибки в употреблении скобок.
8. Методы диагностики и исправления ошибок. Контекстно-зависимые ошибки.
9. Методы диагностики и исправления ошибок. Ошибки, связанные с употреблением различных типов данных.
10. Методы диагностики и исправления ошибок. ошибки выполнения: нахождение индекса массива вне области действия, целочисленное переполнение, попытка чтения за пределами файла.
11. Методы диагностики и исправления ошибок: ошибки, связанные с нарушением ограничений на размер программ, число элементов в таблице символов, размер стека разбора и пр.
12. Семантические действия.
13. Создание надежных компиляторов. Использование формального определения.
14. Создание надежных компиляторов. Модульное проектирование.
15. Создание надежных компиляторов. Проверка компилятора.
16. Анализатор, проверяющий принадлежность грамматики к LL(1)-типу.
17. Программа для построения таблиц синтаксического LL-разбора.
18. Программа для построения таблиц синтаксического LR-разбора.
19. Синтаксический анализатор на основе грамматики простого предшествования.
20. Синтаксический анализатор на основе грамматики расширенного предшествования.
21. Синтаксический анализатор на основе грамматики операторного предшествования.
22. Схемы управления памятью: статическое управление памятью.
23. Схемы управления памятью: стековое управление памятью.

Паспорт лабораторных работ

по дисциплине «Теория языков программирования и методы трансляции», 6 семестр

1. Методика оценки

Лабораторные работы выполняются по варианту, назначенному преподавателем. При выполнении лабораторной работы студент должен ознакомиться с методикой выполнения лабораторной работы, выполнить все пункты работы, подготовить отчет и защитить работу.

Защита происходит в устной форме в виде ответа на один контрольный вопрос.

Варианты лабораторной работы, пункты выполнения лабораторной работы и контрольные вопросы приведены в методических указаниях к лабораторным работам.

2. Критерии оценки

- Лабораторная работа считается выполнена **неудовлетворительно**, если написанная студентом программа не выполняет действия, обусловленные вариантом задания, студент при ответе на вопросы не дает определений основных понятий, оценка составляет *0-7 баллов*.
- Лабораторная работа считается выполнена на **пороговом** уровне, если написанная студентом программа выполняет основные действия, обусловленные вариантом задания, студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, оценка составляет *8-10 баллов*.
- Лабораторная работа считается выполнена на **базовом** уровне, если написанная студентом программа выполняет все действия, обусловленные вариантом задания, студент при ответе на вопросы формулирует основные понятия, законы, дает характеристику процессов, явлений, проводит анализ причин, условий, может представить качественные характеристики процессов, оценка составляет *11-13 баллов*.
- Лабораторная работа считается выполнена на **продвинутом** уровне, если написанная студентом программа выполняет все действия, обусловленные вариантом задания, студент реализовал также дополнительные возможности программы по интерфейсу и/или диагностике и исправлению ошибок, студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, не допускает ошибок и способен обосновать выбор метода решения задачи, оценка составляет *14-16 баллов*.

3. Шкала оценки

Лабораторная работа считается выполнена, если оценена не менее чем на 8 баллов (из 16 возможных).

4. Вопросы к защите лабораторных работ по дисциплине «Теория языков программирования и методы трансляции»

Контрольные вопросы к лабораторной работе №1
ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ТАБЛИЦ, ИСПОЛЬЗУЕМЫХ В ТРАНСЛЯТОРЕ

1. Постоянные таблицы, используемые в трансляторах.
2. Переменные таблицы, используемые в трансляторах.
3. Операции с таблицами.
4. Способы организации таблиц.
5. Хеш-функции.
6. Методы рехеширования.
7. Таблицы с вычисляемым входом.

Контрольные вопросы к лабораторной работе №2 РАЗРАБОТКА И РЕАЛИЗАЦИЯ БЛОКА ЛЕКСИЧЕСКОГО АНАЛИЗА

1. Интерпретаторы и компиляторы.
2. Основные фазы трансляции.
3. Регулярные выражения и конечные автоматы.
4. Детерминированные конечные автоматы.
5. Методы приведения недетерминированного конечного автомата к детерминированному.
6. Автоматная грамматика.
7. Диаграмма состояний сканера.

Контрольные вопросы к лабораторной работе №3 РАЗРАБОТКА И РЕАЛИЗАЦИЯ БЛОКА СИНТАКСИЧЕСКОГО АНАЛИЗА

1. Нисходящая стратегия разбора.
2. Восходящая стратегия разбора.
3. Синтаксические деревья.
4. Классификация языков по Хомскому.
5. *LL*-грамматика.
6. *LL*-таблица разбора.
7. *LR*-грамматика.
8. *LR*-таблица разбора.

Контрольные вопросы к лабораторной работе №4 РАЗРАБОТКА И РЕАЛИЗАЦИЯ БЛОКА ГЕНЕРАЦИИ КОДА

1. Внутренние формы представления программы.
2. Постфиксная запись.
3. Представление генерируемого кода в форме четверок.
4. Организация генератора кода.
5. Среда выполнения.
6. Методы управления памятью.
7. Фазы управления памятью.

5. Задания на лабораторные работы по дисциплине «Теория языков программирования и методы трансляции»

Задание на лабораторную работу состоит в разработке и реализации транслятора программы, написанной на подмножестве языка Си++ (вид подмножества указан в варианте задания) в программу на языке Ассемблер.

ВАРИАНТЫ ЗАДАНИЙ К ЛАБОРАТОРНЫМ РАБОТАМ

- 1) Подмножество языка C++ включает:
 - данные типа **int**;
 - инструкции описания переменных;
 - операторы присваивания, **if**, **if- else** любой вложенности и в любой последовательности;
 - операции **+**, **-**, *****, **=**, **!=**, **<**.

- 2) Подмножество языка C++ включает:
 - данные типа **int**;
 - инструкции описания переменных;
 - операторы присваивания, **while** любой вложенности и в любой последовательности;
 - операции $+$, $-$, $<=$, $>=$, $<$, $>$.
- 3) Подмножество языка C++ включает:
 - данные типа **int**;
 - инструкции описания переменных;
 - операторы присваивания, **do-while** любой вложенности и в любой последовательности;
 - операции $+=$, $-=$, $+$, $-$, $==$, $!=$.
- 4) Подмножество языка C++ включает:
 - данные типа **int**;
 - инструкции описания переменных;
 - операторы присваивания, **for** любой вложенности и в любой последовательности;
 - операции $+$, $-$, $*$, $\&\&$, $\|$.
- 5) Подмножество языка C++ включает:
 - данные типа **int**;
 - инструкции описания переменных;
 - операторы присваивания, **switch** любой вложенности и в любой последовательности;
 - операции $+$, $-$, $*$, $=$, $!=$, $<$.
- 6) Подмножество языка C++ включает:
 - данные типа **int, float, char**;
 - инструкции описания переменных;
 - операторы присваивания в любой последовательности;
 - операции $+$, $-$, $*$, $=$, $!=$, $<$, $>$.
- 7) Подмножество языка C++ включает:
 - данные типа **int, float, массивы** из элементов указанных типов;
 - инструкции описания переменных;
 - операторы присваивания в любой последовательности;
 - операции $+$, $-$, $*$, $=$, $!=$, $<$, $>$.
- 8) Подмножество языка C++ включает:
 - данные типа **int, float, struct** из элементов указанных типов;
 - инструкции описания переменных;
 - операторы присваивания в любой последовательности;
 - операции $+$, $-$, $*$, $=$, $!=$, $<$, $>$.
- 9) Подмножество языка C++ включает:
 - данные типа **int, char**;
 - инструкции описания переменных;
 - операторы присваивания в любой последовательности;
 - операции $+$, $-$, $<$, $>$, побитовые операции $<<$, $>>$, $\&$, $|$.
- 10) Подмножество языка C++ включает:

- данные типа **int**;
- инструкции описания переменных;
- операторы присваивания в любой последовательности;
- полный набор арифметических, логических операций и операций сравнения.

11) Подмножество языка C++ включает:

- данные типа **int**;
- инструкции описания переменных и функций;
- несколько функций с параметрами и возвращаемым значением;
- операторы присваивания в любой последовательности;
- операции $+$, $-$, $=$, $!=$, $<$, $>$.