



Министерство здравоохранения Российской Федерации

**федеральное государственное бюджетное образовательное учреждение
высшего образования**

"Северо-Западный государственный медицинский университет имени И.И. Мечникова"

Министерства здравоохранения Российской Федерации

(ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Основы программирования»

Специальность: 30.05.03 Медицинская кибернетика

Направленность: Цифровые технологии медицины и здравоохранения

2024

Подлинник электронного документа, подписанного ЭП, хранится в системе электронного документооборота ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России	
СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП	
Сертификат	1FC08DD37C5678CF72030C7355B41753
Владелец	Сайганов Сергей Анатольевич
Действителен	с 22.10.2024 14:51:43 по 15.01.2026 14:51:43

Рабочая программа дисциплины «Основы программирования» составлена на основании Федерального государственного образовательного стандарта высшего образования – специалитет по специальности 30.05.03 Медицинская кибернетика, утвержденного приказом Министерства науки и высшего образования Российской Федерации от 13 августа 2020 года № 1006 «Об утверждении федерального государственного образовательного стандарта высшего образования - специалитет по специальности 30.05.03 Медицинская кибернетика»

Составители рабочей программы дисциплины:

Расказова Альбина Николаевна, кандидат технических наук, доцент, доцент кафедры медицинской информатики и физики; Шматко Алексей Дмитриевич, доктор экономических наук, профессор, заведующий кафедрой медицинской информатики и физики

Рецензент:

Иванов Алексей Сергеевич – доцент кафедры Федерального государственного бюджетного образовательного учреждения высшего образования «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича», канд.техн.наук.

Рассмотрено Методическим советом и рекомендовано для утверждения на Ученом совете 22 ноября 2024 г.

Председатель _____ /Артюшкин С.А./

Дата обновления:

СОДЕРЖАНИЕ

1. Цель освоения дисциплины	4
2. Место дисциплины в структуре образовательной программы.....	4
3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы	4
4. Объем дисциплины и виды учебной работы.....	4
5. Содержание дисциплины, структурированное по разделам (темам) с указанием количества академических часов и видов учебных занятий	5
6. Методические указания для обучающихся по освоению дисциплины	8
7. Оценочные материалы.....	8
8. Перечень учебной литературы и ресурсов сети «Интернет», необходимых для освоения дисциплины.....	8
9. Перечень информационных технологий, используемых для освоения дисциплины, включая перечень программного обеспечения, профессиональных баз данных и информационных справочных систем	8
10. Материально-техническое обеспечение дисциплины.....	11
Приложение А	12

1. Цель освоения дисциплины

Целью освоения дисциплины «Основы программирования» является формирование компетенций обучающегося, необходимых для разработки и внедрения программных решений, направленных на улучшение диагностики, лечения и управления медицинскими данными в сфере здравоохранения, а также фундаментальных знаний и навыков в области алгоритмизации, освоения языков программирования, развития логического мышления и критического анализа, знакомства с современными инструментами разработки и принципами объектно-ориентированного программирования.

2. Место дисциплины в структуре образовательной программы

Дисциплина «Основы программирования» относится к обязательной части Блока 1 «Дисциплины (модули)» основной профессиональной образовательной программы по специальности 30.05.03 Медицинская кибернетика (уровень образования специалитет), направленность: Цифровые технологии медицины и здравоохранения. Дисциплина является обязательной к изучению.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
ОПК-7. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ИД-1 ОПК-7.1. Разрабатывает алгоритмы и компьютерные программы для решения прикладных задач в области профессиональной деятельности.
	ИД-2 ОПК-7.2. Проводит отладку и тестирование компьютерных программ для решения прикладных задач в области профессиональной деятельности.
ПК-3. Ведет медицинскую документацию и контролирует качество ее ведения с помощью информационных систем в сфере здравоохранения	ИД-2 ПК-3.2. Разрабатывает формы медицинской документации на основе кодирования и представления данных в соответствии с нормативными, методическими требованиями и перспективами автоматизированной обработки
ПК-5. Осуществляет системное администрирование и сопровождение информационных систем и технологий медицинской организации	ИД-2 ПК-5.4. Разрабатывает программы применения интеллектуальных систем для решения профессиональных задач работников медицинской организации

Код индикатора достижения компетенции	Результаты обучения (показатели оценивания)	Оценочные средства
ИД-1 ОПК-7.1.	знает основы синтаксиса языков программирования,	Тестовые задания

	<p>типы и структуры данных, принципы алгоритмизации, основы объектно-ориентированного программирования и процессы разработки программного обеспечения</p>	
	<p>умеет писать программы, решать задачи с использованием алгоритмов, применять средства разработки, читать и понимать код, а также работать с ошибками</p>	Тестовые задания
	<p>имеет навык структурировать информацию и строить логические цепочки, анализировать задачи и находить оптимальные пути их решения</p>	Типовые задачи
ИД-2 ОПК-7.2.	<p>знает методы и техники отладки и тестирования программного обеспечения, основные виды тестирования, принципы работы с инструментами для автоматизации тестирования</p>	Тестовые задания
	<p>умеет разрабатывать и выполнять тестовые сценарии, выявлять и анализировать ошибки в программном обеспечении, использовать инструменты для отладки и тестирования</p>	Тестовые задания
	<p>имеет навык проводить тестирование программ, а также документировать результаты тестирования и отладки</p>	Типовые задачи
ИД-2 ПК-3.2.	<p>знает требования к интерфейсу форм медицинской документации, основы кодирования и представления данных, а также современные подходы к автоматизированной обработке информации в здравоохранении</p>	Тестовые задания
	<p>умеет разрабатывать формы медицинской документации, применять методы кодирования данных, адаптировать документацию для автоматизации обработки данных.</p>	Тестовые задания
	<p>имеет навык критического и логического мышления, командной работы, самостоятельного обучения и документирования кода</p>	Типовые задачи
ИД-2 ПК-5.4.	<p>знает методы обработки и анализа медицинских данных, основные концепции и методы разработки с помощью интеллектуальных систем, а также специфику применения технологий искусственного интеллекта</p>	Тестовые задания
	<p>умеет разрабатывать и внедрять программы, использующие интеллектуальные системы для решения профессиональных задач, анализировать данные, создавать модели для поддержки принятия решений и оптимизации процессов в медицинских организациях</p>	Типовые задачи
	<p>имеет навык разработки программного кода, в том числе для осуществления системного администрирования, а также навыки аналитического мышления и внимательности к деталям</p>	Типовые задачи

4. Объем дисциплины и виды учебной работы

Вид учебной работы	Трудоемкость	Семестры	
		5	6
Контактная работа обучающихся с преподавателем:	106	52	54
Лекции	32	16	16
Практические занятия	72	36	36
Семинары	-	-	-
Промежуточная аттестация: зачет, в том числе сдача и групповые консультации	2	-	2
Самостоятельная работа:	38	20	18
в период теоретического обучения	34	20	14
подготовка к сдаче зачета	4	-	4
Общая трудоемкость:	академических часов	144	72
	зачетных единиц	4	2

5. Содержание дисциплины, структурированное по разделам (темам) с указанием количества академических часов и видов учебных занятий

5.1. Содержание разделов дисциплины

№ п/п	Наименование раздела дисциплины	Аннотированное содержание раздела дисциплины	Перечень компетенций, формируемых в процессе освоения раздела
1	Введение в программирование	Цели и задачи дисциплины. Этапы решения задач на ЭВМ. Понятие программы, программирования, виды программирования. Языки программирования, классификация. Трансляция программ. Эволюция языков программирования. Трансляторы, интерпретаторы, компиляторы, компоновщики.	ОПК-7 ПК-3 ПК-5
2	Основы языка C++	Введение в язык программирования C++, особенности языка. Основные понятия: переменные, типы данных, операторы, выражения. Управление потоком выполнения программы: условные операторы, циклы. Работа с массивами и строками. Функции: объявление, параметры, возвращаемые значения, рекурсия. Препроцессорные директивы. Ввод/вывод данных. Указатели: объявление, арифметика указателей, работа с динамической памятью (new, delete). Массивы и указатели. Строки и указатели. Утечки памяти и их предотвращение.	ОПК-7 ПК-3 ПК-5

3	Объектно-ориентированное программирование	Классы и объекты: объявление, члены класса, конструкторы, деструкторы, методы. Инкапсуляция, наследование, полиморфизм. Виртуальные функции и абстрактные классы. Шаблоны функций и классов. Исключения и обработка ошибок. Отладка и тестирование программного кода	ОПК-7 ПК-3 ПК-5
4	Стандартная библиотека шаблонов	Контейнеры STL: vector, list, map, set, deque. Итераторы. Алгоритмы STL: sort, find, copy, transform. Функторы, адаптеры. Создание собственных библиотек. Системы управления версиями (Git)	ОПК-7 ПК-3 ПК-5

5.2. Тематический план лекций

№ п/п	Наименование раздела дисциплины	Тематика лекций	Активные формы обучения	Трудоемкость (академических часов)
1	Введение в программирование	Л.1. Введение в C++. Основные понятия.	ЛБ	2
		Л.2. Управление потоком выполнения. Ввод/вывод.	ЛБ	2
		Л.3. Массивы и строки.	ЛБ	2
		Л.4. Функции. Препроцессор.	ЛБ	2
2	Основы языка C++	Л.5. Указатели. Основные понятия.	ЛБ	2
		Л.6. Динамическая память. Управление памятью.	ЛБ	2
3	Объектно-ориентированное программирование	Л.7. Классы и объекты. Основы ООП.	ЛБ	2
		Л.8. Конструкторы и деструкторы. Перегрузка операторов.	ЛБ	2
		Л.9. Наследование. Полиморфизм.	ЛБ	2
		Л.10. Абстрактные классы. Интерфейсы.	ЛБ	2
		Л.11. Шаблоны функций и классов.	ЛБ	2
		Л.12. Обработка исключений.	ЛБ	2
4	Стандартная библиотека шаблонов	Л.13. Контейнеры STL: vector, list, deque map, set и другие.	ЛБ	2
		Л.14. Итераторы и алгоритмы STL.	ЛБ	2

		Л.15. Функторы и лямбда-выражения	ЛБ	2
		Л.16. Интеграция C++ и Unreal Engine	ЛБ	2
				32

ЛБ – лекция-беседа

5.3. Тематический план практических занятий

№ п/п	Наименование раздела дисциплины	Тематика практических занятий	Активные формы обучения	Формы текущего контроля	Трудоемкость (академических часов)
1	Введение в программирование	ПЗ.1. Типы данных и операторы	ИТ	тестирование	4
		ПЗ.2. Условные операторы и циклы	ИТ	решение типовых задачи	4
		ПЗ.3. Массивы	ИТ	решение типовых задачи	4
		ПЗ.4. Функции и препроцессор	ИТ	тестирование	4
2	Основы языка C++	ПЗ.5. Указатели	ИТ	тестирование	4
		ПЗ.6. Динамическая память	ИТ	тестирование	4
3	Объектно-ориентированное программирование	ПЗ.7. Классы и объекты	ИТ	тестирование	4
		ПЗ.8. Наследование	ИТ	решение типовых задачи	4
		ПЗ.9. Виртуальные функции и полиморфизм	ИТ	решение типовых задачи	4
		ПЗ.10. Перегрузка операторов	ИТ	тестирование	4
		ПЗ.11. Шаблоны	ИТ	решение типовых задачи	4
4	Стандартная библиотека шаблонов	ПЗ.12. Обработка исключений	ИТ	тестирование	4
		ПЗ.13. Контейнеры STL	ИТ	решение типовых задачи	4

		ПЗ.14. Контейнеры STL	ИТ	решение типовых задачи	4
		ПЗ.15. Алгоритмы и итераторы STL	ИТ	решение типовых задачи	4
		ПЗ.16. Алгоритмы и итераторы STL	ИТ	решение типовых задачи	4
		ПЗ.17. Unreal Engine	ИТ	решение типовых задачи	4
		ПЗ.18. Unreal Engine	ИТ	решение типовых задачи	4
					72

ИТ - имитационной тренинг

АИМ - анализ имитационных моделей

5.4. Самостоятельная работа:

№ п/п	Наименование раздела дисциплины	Виды самостоятельной работы	Формы текущего контроля	Трудоемкость (академических часов)
1	Введение в программирование	<i>Работа с лекционным материалом</i>	тестирование	8
2	Основы языка C++	<i>Работа с лекционным материалом</i>	тестирование	8
3	Объектно-ориентированное программирование	<i>Работа с учебной литературой</i>	решение типовых задачи	8
4	Стандартная библиотека шаблонов	<i>Работа с учебной литературой</i>	решение типовых задачи	10
		<i>Подготовка к сдаче зачета</i>		4
ИТОГО:				38

5.5. Перечень нормативных документов:

ГОСТ Р ИСО/МЭК 90003-2014 Разработка программных продуктов. Руководящие указания по применению ИСО 9001:2008 при разработке программных продуктов.

ГОСТ Р 51904-2002 Программное обеспечение встроенных систем. Общие требования к разработке и документированию

ГОСТ 19.701-90 (ИСО 5807-85) Единая система программной документации.
Схемы алгоритмов, программ, данных и систем

6. Методические указания для обучающихся по освоению дисциплины

Для успешного освоения учебной дисциплины обучающемуся рекомендуется посещать лекционные и практические занятия в соответствии с расписанием учебных занятий, своевременно и в полном объеме выполнять задания текущего контроля, пройти промежуточную аттестацию.

Подготовка к лекциям

Лекции по дисциплине проводятся в традиционной и интерактивной форме с использованием технических средств обучения. Во время лекций студенту необходимо вести конспект лекции, структура и объем которого определяется самостоятельно. Основой формирования конспекта являются аудио, видеоматериалы, презентации лектора по тематике лекции, а также рекомендованная учебная литература, ресурсы сети «Интернет» и/или нормативные документы.

Подготовка к практическим занятиям и выполнение практических занятий

Практические занятия проводятся с использованием активных форм обучения. При подготовке к практическим занятиям необходимо выполнять задания для самостоятельной работы. В программе дисциплины предусмотрены мероприятия текущего контроля для проверки освоения разделов дисциплины в рамках самостоятельной работы. Контроль выполнения заданий на практических занятиях осуществляется с помощью мероприятий текущего контроля.

Рекомендации по работе с литературой

В программе дисциплины представлен список литературы и ресурсов сети «Интернет», необходимых для освоения дисциплины.

Рекомендации по подготовке к текущему контролю

С целью контроля освоения дисциплины в тематическом плане занятий предусмотрены контрольные мероприятия, которые составляют средства текущего контроля. В рабочей программе дисциплины текущий контроль представлен тестовыми заданиями и решением типовых задач.

Рекомендации по подготовке к промежуточной аттестации

Промежуточная аттестация по дисциплине проводится в форме зачета. Зачет включает в себя: собеседование по контрольным вопросам.

7. Оценочные материалы

Оценочные материалы по дисциплине для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся включают в себя примеры оценочных средств (Приложение А к рабочей программе дисциплины), процедуру и критерии оценивания.

8. Перечень учебной литературы и ресурсов сети «Интернет», необходимых для освоения дисциплины

8.1. Учебная литература:

1. Абдулаева, З. И. Информационные компьютерные системы в медицине и здравоохранении : Учеб.-метод. пособие / З. И. Абдулаева, А. Д. Шматко; М-во здравоохранения Рос. Федерации, ФГБОУ ВО Сев.-Зап. гос. мед. ун-т им. И. И. Мечникова, Каф. мед. информатики и физики. - СПб. : Изд-во СЗГМУ им. И. И. Мечникова, 2017. - 43 с. : рис. - (Медицинское образование). - Библиогр.: с. 43 (11 назв.). https://sdo.szgmu.ru/pluginfile.php/178023/mod_resource/content/1/ Абдулаева З. И., Шматко А. Д. Информационные компьютерные системы в медицине и здравоохранении.pdf

2. Абдулаева, З. И. Медицинская информатика [Текст] : учебное пособие : [в 2 ч.] / З. И. Абдулаева, Д. Ф. Курбанбаева. Теоретические основы медицинской информатики. — Санкт-Петербург : Изд-во Политехнического ун-та, 2018-, 2018. — 190 с. : ил., табл.; ISBN 978-5-7422-6240-4. – режим доступа: https://rusneb.ru/catalog/000199_000009_009794428/

3. Медицинская информатика : учебник / Т. В. Зарубина, Б. А. Кобринский, С. С. Белоносов [и др.]; ред. Т. В. Зарубина, Б. А. Кобринский. - М. : ГЭОТАР-Медиа, 2018. - 507 с. : цв. ил., табл. - Терминол. словарь: с. 490-491.- Библиогр.: с. 500-501.- Предм. указ.: с. 502-507. - ISBN 978-5-9704-4573-0.

4. Принципы функционирования интеллектуальной системы динамического контроля факторов риска и формирования рекомендаций по здоровьесбережению / Б. А. Кобринский, А. С. Кадыков, М. Г. Полтавская [и др.] // Профилактическая медицина. - 2019. - Т. 22, № 5. - С. 78-84. - Библиогр.: 26 назв. Ресурсы информационно-телекоммуникационной сети «Интернет»:

8.2. Ресурсы информационно-телекоммуникационной сети «Интернет»:

Наименования ресурса сети «Интернет»	Электронный адрес ресурса
Документация <i>Pandas</i>	https://pandas.pydata.org/docs/
Документация <i>Python 3.13.0</i>	https://docs.python.org/3/
Документация <i>NumPy</i>	https://numpy.org/doc/

9. Перечень информационных технологий, используемых для освоения дисциплины, включая перечень программного обеспечения, профессиональных баз данных и информационных справочных систем

9.1. Перечень информационных технологий, используемых при осуществлении образовательного процесса:

№ п/п	Наименование раздела дисциплины	Информационные технологии
1	Введение в программирование	Контроль знаний - тестирование в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России, Размещение учебных материалов в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России
2	Основы языка C++	Контроль знаний - тестирование в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России, Размещение учебных материалов в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России
3	Объектно-ориентированное программирование	Контроль знаний - тестирование в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России, Размещение учебных материалов в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России

4	Стандартная библиотека шаблонов	Контроль знаний - тестирование в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России, Размещение учебных материалов в ЭИОС ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России
---	---------------------------------	---

9.2. Перечень программного обеспечения, используемого при осуществлении образовательного процесса (лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства):

№ п/п	Наименование программного продукта	Срок действия лицензии	Документы, подтверждающие право использования программных продуктов
лицензионное программное обеспечение			
1.	Dr. Web	1 год	Контракт № 265-2023-ЗК
2.	MS Windows 8 MS Windows 8.1 MS Windows 10 MS Windows Server 2012 Datacenter - 2 Proc MS Windows Server 2012 R2 Datacenter - 2 Proc MS Windows Server 2016 Datacenter Core	Неограниченно	Государственный контракт № 30/2013-О; Государственный контракт № 399/2013-ОА; Государственный контракт № 07/2017-ЭА.
3.	MS Office 2010 MS Office 2013	Неограниченно	Государственный контракт № 30/2013-ОА; Государственный контракт № 399/2013-ОА.
4.	Academic LabVIEW Premium Suite (1 User)	Неограниченно	Государственный контракт № 02/2015
лицензионное программное обеспечение отечественного производства			
1.	Антиплагиат	1 год	Договор № 133/2024-М
2.	«WEBINAR (ВЕБИНАР)» ВЕРСИЯ 3.0	1 год	Контракт № 211/2024-ЭА
3.	«Среда электронного обучения ЗКЛ»	1 год	Контракт № 121/2024-ЗЗЕП
4.	TrueConf Enterprise	1 год	Контракт № 216/2024-ЭА
свободно распространяемое программное обеспечение			
1.	Google Chrome	Неограниченно	Открытое лицензионное соглашение GNU GeneralPublicLicense
2.	NVDA	Неограниченно	Открытое лицензионное соглашение GNU GeneralPublicLicense
3.	MS Visual Studio учебная версия	Неограниченно	Открытое лицензионное соглашение GNU GeneralPublicLicense
свободно распространяемое программное обеспечение отечественного производства			
1.	Moodle	Неограниченно	Открытое лицензионное соглашение GNU GeneralPublicLicense

9.3. Перечень профессиональных баз данных и информационных справочных систем:

№ п/п	Наименование программного продукта	Срок действия лицензии	Документы, подтверждающие право использования программных продуктов	Режим доступа для обучающихся – инвалидов и лиц с ограниченными возможностями здоровья
1.	Консультант Плюс	1 год	Контракт № 1067/2021-ЭА	-
2.	ЭБС «Консультант студента»	1 год	Контракт № 97/2023-ЭА	https://www.studentlibrary.ru/
3.	ЭБС «Айбукс.ру/ibooks.ru»	1 год	Договор № 207/2023-ЗЗЕП	https://ibooks.ru
4.	Цифровой образовательный ресурс IPRsmart	1 год	Договор № 206/2023-ЗЗЕП	http://www.iprbookshop.ru/
5.	Электронно-библиотечная система «Букап»	1 год	Договор № 199/2023-ЗЗЕП	https://www.books-up.ru/
6.	ЭБС «Издательство Лань»	1 год	Договор № 200/2023-ЗЗЕП	https://e.lanbook.com/
7.	Образовательная платформа ЮРАЙТ	1 год	Договор № 155/2023-ПЗ	https://urait.ru/
8.	Электронные издания в составе базы данных НАУЧНАЯ ЭЛЕКТРОННАЯ БИБЛИОТЕКА eLIBRARY.RU	1 год	Лицензионный договор № SU-7139/2024	https://www.elibrary.ru/defaultx.asp
9.	Программное обеспечение «Платформа mb4» в части Справочно-информационной системы «MedBaseGeotar»	1 год	Лицензионный договор № 97/2024-ЗЗЕП	https://mbasegeotar.ru/
10.	Универсальные базы электронных периодических изданий ИВИС	1 год	Лицензионный договор № 116/2023-ЗЗЕП «Журналы России по медицине и здравоохранению» Лицензионный договор № 42/2023-ЗЗЕП «Индивидуальные издания»	https://dlib.eastview.com/
11.	Создание Виртуального читального зала Российской государственной библиотеки (ВЧЗ РГБ) для обслуживания удаленного пользователя	1 год	Лицензионный договор № 120/2024-М14	https://search.rsl.ru/

10. Материально-техническое обеспечение дисциплины

Учебные аудитории для проведения учебных занятий лекционного типа (в соответствии со справкой о материально-техническом обеспечении).

Учебная аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля успеваемости и промежуточной аттестации обучающихся (в соответствии со справкой о материально-техническом обеспечении).

Помещение для самостоятельной работы обучающихся: компьютерная техника с возможностью подключения к сети «Интернет», обеспеченная доступом к электронной информационно-образовательной среде ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России (в соответствии со справкой о материально-техническом обеспечении).

Министерство здравоохранения Российской Федерации

**федеральное государственное бюджетное образовательное учреждение
высшего образования**

**«Северо-Западный государственный медицинский университет
имени И.И. Мечникова»**

Министерства здравоохранения Российской Федерации

(ФГБОУ ВО СЗГМУ им. И.И. Мечникова Минздрава России)

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

(для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся)

Специальность:	30.05.03 Медицинская кибернетика
Направленность:	Цифровые технологии медицины и здравоохранения
Наименование дисциплины:	Основы программирования

1. Перечень планируемых результатов обучения по дисциплине

Код индикатора достижения компетенции	Результаты обучения (показатели оценивания)	Оценочные средства
ИД-1 ОПК-7.1.	знает основы синтаксиса языков программирования, типы и структуры данных, принципы алгоритмизации, основы объектно-ориентированного программирования и процессы разработки программного обеспечения	Тестовые задания
	умеет писать программы, решать задачи с использованием алгоритмов, применять средства разработки, читать и понимать код, а также работать с ошибками	Тестовые задания
	имеет навык структурировать информацию и строить логические цепочки, анализировать задачи и находить оптимальные пути их решения	Типовые задачи
ИД-2 ОПК-7.2.	знает методы и техники отладки и тестирования программного обеспечения, основные виды тестирования, принципы работы с инструментами для автоматизации тестирования	Тестовые задания
	умеет разрабатывать и выполнять тестовые сценарии, выявлять и анализировать ошибки в программном обеспечении, использовать инструменты для отладки и тестирования	Тестовые задания
	имеет навык проводить тестирование программ, а также документировать результаты тестирования и отладки	Типовые задачи
ИД-2 ПК-3.2.	знает требования к интерфейсу форм медицинской документации, основы кодирования и представления данных, а также современные подходы к автоматизированной обработке информации в здравоохранении	Тестовые задания
	умеет разрабатывать формы медицинской документации, применять методы кодирования данных, адаптировать документацию для автоматизации обработки данных.	Тестовые задания
	имеет навык критического и логического мышления, командной работы, самостоятельного обучения и документирования кода	Типовые задачи
ИД-2 ПК-5.4.	знает методы обработки и анализа медицинских данных, основные концепции и методы разработки с помощью интеллектуальных систем, а также специфику применения технологий искусственного интеллекта	Тестовые задания
	умеет разрабатывать и внедрять программы, использующие интеллектуальные системы для решения профессиональных задач, анализировать данные, создавать модели для поддержки принятия решений и оптимизации процессов в медицинских организациях	Типовые задачи
	имеет навык разработки программного кода, в том числе для осуществления системного администрирования, а также навыки аналитического мышления и внимательности к деталям	Типовые задачи

2. Примеры оценочных средств и критерии оценивания для проведения текущего контроля

2.1. Примеры входного контроля

Что такое алгоритм?

- 1) Набор данных

- 2) **Последовательность действий для решения задачи**
- 3) Язык программирования
- 4) Компьютерная программа

Какой из следующих методов является методом "разделяй и властвуй"?

- 1) Сортировка вставками
- 2) **Быстрая сортировка**
- 3) Сортировка выбором
- 4) Линейный поиск

Критерии оценки, шкала оценивания тестовых заданий

Оценка	Описание
«зачтено»	Выполнено с отклонением – 85%-100%
«зачтено»	Выполнено с отклонением – 70%-84%
«незачтено»	Выполнено частично – 69% и менее правильных ответов

2.2. Примеры тестовых заданий

ИД-1 ОПК-7.1.

Вопрос № 1

Какой оператор используется для сравнения двух значений на равенство в большинстве языков программирования?

- 1) =
- 2) **==**
- 3) ===
- 4) !=

ИД-2 ОПК-7.2.

Вопрос № 2

Какой из следующих терминов относится к объектно-ориентированному программированию?

- 1) Функция
- 2) **Класс**
- 3) Процедура
- 4) Алгоритм

ИД-2 ПК-3.2.

Вопрос № 3

Какой из следующих циклов используется для выполнения блока кода определенное количество раз?

- 1) while
- 2) **for**
- 3) do-while
- 4) repeat

ИД-2 ПК-5.4.

Вопрос № 4

Что такое цикл в программировании?

- 1) Структура, которая выполняет блок кода один раз
- 2) **Структура, которая выполняет блок кода несколько раз**
- 3) Структура, которая завершает выполнение программы
- 4) Структура, которая обрабатывает ошибки

Критерии оценки, шкала оценивания тестовых заданий

Оценка	Описание
«отлично»	Выполнено в полном объеме – 90%-100%
«хорошо»	Выполнено не в полном объеме – 80%-89%
«удовлетворительно»	Выполнено с отклонением – 70%-79%
«неудовлетворительно»	Выполнено частично – 69% и менее правильных ответов

2.3. Примеры типовых задач

ИД-1 ОПК-7.1.

Вычисление факториала

Задача 1:

Напишите функцию `factorial`, которая вычисляет факториал целого неотрицательного числа. Факториал числа n (обозначается $n!$) определяется как произведение всех целых чисел от 1 до n . Например, $5! = 1 * 2 * 3 * 4 * 5 = 120$. Функция должна возвращать вычисленный факториал. Обработайте случай, когда введенное число отрицательное (например, выведите сообщение об ошибке или верните -1). В основной функции запросите у пользователя число и выведите его факториал.

Задача 2: Поиск максимального элемента в массиве

Напишите функцию `findMax`, которая принимает массив целых чисел и его размер в качестве аргументов и возвращает значение максимального элемента в массиве. В основной функции создайте массив целых чисел, инициализируйте его значениями и вызовите функцию `findMax` для нахождения максимального элемента. Выведите результат на экран.

Задача 3: Обработка строки

С использованием стандартной библиотеки строк (`string`), напишите программу, которая запрашивает у пользователя строку и выполняет следующие действия:

Подсчитывает количество гласных букв (а, о, е, и, э, ю, я) в строке.

Преобразует строку в верхний регистр.

Выводит исходную строку, количество гласных и преобразованную строку в верхний регистр.

ИД-2 ОПК-7.2.

Задача 4: Отладка программы с ошибкой в обработке массива

Вам дан следующий фрагмент кода, содержащий ошибку:

```
#include <iostream>
```

```

int main() {
    int arr[5] = {10, 20, 30, 40, 50};
    for (int i = 0; i <= 5; i++) { // Ошибка в условии цикла
        std::cout << arr[i] << " ";
    }
    std::cout << std::endl;
    return 0;
}

```

Этот код пытается вывести элементы массива arr. Однако в нем присутствует ошибка, которая приводит к выходу за границы массива.

Найдите ошибку: Определите, в чем заключается ошибка в коде.

Исправьте ошибку: Исправьте код так, чтобы он корректно выводил элементы массива.

Проверьте исправление: Запустите исправленный код и убедитесь, что он работает правильно.

Добавьте обработку ошибок: Модифицируйте код так, чтобы он обрабатывал ситуацию, когда размер массива неизвестен заранее (например, массив передается в функцию). Добавьте проверку на выход за границы массива и выведите сообщение об ошибке в случае возникновения такой ситуации.

Задача 5. Обработка исключений в функции

Напишите функцию divide, которая делит два целых числа. Функция должна обрабатывать исключение std::runtime_error, если делитель равен нулю. Если деление выполнено успешно, функция должна вернуть результат.

```

#include <iostream>
#include <stdexcept>
double divide(int a, int b) {
    // Добавьте обработку исключений здесь
    if (b == 0) {
        throw std::runtime_error("Деление на ноль!");
    }
    return (double)a / b;
}
int main() {
    try {
        std::cout << divide(10, 2) << std::endl;
    }
}

```

```

    std::cout << divide(10, 0) << std::endl;
} catch (const std::runtime_error& error) {
    std::cerr << "Ошибка: " << error.what() << std::endl;
}
return 0;
}

```

ИД-2 ПК-3.2.

Задача 6. Дополнение текстового интерфейса для управления списком задач

Дан фрагмент кода для текстового интерфейса, который позволяет пользователю управлять списком задач. Список задач хранится в векторе строк. Необходимо дополнить код функциями для добавления, удаления и вывода списка задач.

```

#include <iostream>

#include <vector>

#include <string>

using namespace std;

vector<string> tasks;

void displayMenu() {
    cout << "Меню:" << endl;
    cout << "1. Добавить задачу" << endl;
    cout << "2. Удалить задачу" << endl;
    cout << "3. Показать задачи" << endl;
    cout << "4. Выход" << endl;
    cout << "Ваш выбор: ";
}

// Добавьте функцию addTask, которая добавляет задачу в tasks

void addTask() {
    string newTask;
    cout << "Введите новую задачу: ";
    getline(cin >> ws, newTask); // Используем getline для корректного ввода строк с пробелами
    tasks.push_back(newTask);
    cout << "Задача добавлена!" << endl;
}

```

// Добавьте функцию deleteTask, которая удаляет задачу из tasks по индексу

```
void deleteTask() {
    if (tasks.empty()) {
        cout << "Список задач пуст." << endl;
        return;
    }
    displayTasks(); // показываем список перед удалением
    int indexToDelete;
    cout << "Введите номер задачи для удаления: ";
    cin >> indexToDelete;
    if (indexToDelete >= 0 && indexToDelete < tasks.size()) {
        tasks.erase(tasks.begin() + indexToDelete);
        cout << "Задача удалена!" << endl;
    } else {
        cout << "Неверный номер задачи." << endl;
    }
}

void displayTasks() {
    if (tasks.empty()) {
        cout << "Список задач пуст." << endl;
        return;
    }
    cout << "Список задач:" << endl;
    for (size_t i = 0; i < tasks.size(); ++i) {
        cout << i << ". " << tasks[i] << endl;
    }
}

int main() {
    int choice;
    do {
        displayMenu();
        cin >> choice;
```

```

cin.ignore(); // очищаем буфер cin после ввода числа
switch (choice) {
    case 1:
        addTask();
        break;
    case 2:
        deleteTask();
        break;
    case 3:
        displayTasks();
        break;
    case 4:
        cout << "Выход..." << endl;
        break;
    default:
        cout << "Неверный выбор." << endl;
}
} while (choice != 4);
return 0;
}

```

ИД-2 ПК-5.4.

Задача 6: Завершение функции сортировки

Дополните функцию bubbleSort, которая сортирует массив целых чисел методом пузырька. Функция должна принимать массив и его размер в качестве аргументов и сортировать массив по возрастанию.

```

#include <iostream>

void bubbleSort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                // Добавьте код здесь для обмена местами arr[j] и arr[j+1]
            }
        }
    }
}

```

```

    }
}

int main() {
    int arr[] = {64, 34, 25, 12, 22, 11, 90};
    int n = sizeof(arr) / sizeof(arr[0]);
    bubbleSort(arr, n);
    for (int i = 0; i < n; i++)
        std::cout << arr[i] << " ";
    std::cout << std::endl;
    return 0;
}

```

Критерии оценки, шкала оценивания по типовым задачам

Оценка	Описание
«зачтено»	Объяснение хода решения задачи подробное, последовательное, грамотное, с правильным и свободным владением терминологией; ответы на дополнительные вопросы верные, четкие
«не зачтено»	Объяснение хода решения задачи подробное, но недостаточно логичное, с единичными ошибками в деталях, ответы на дополнительные вопросы верные, но недостаточно четкие

3. Процедура проведения текущего контроля

Текущий контроль успеваемости по дисциплине проводится в форме тестирования и решения типовых задач

4. Примеры оценочных средств и критерии оценивания для проведения промежуточной аттестации

4.1. Примерный перечень контрольных вопросов для подготовки к экзамену:

ИД-1 ОПК-7.1., ИД-2 ОПК-7.2., ИД-2 ПК-3.2., ИД-2 ПК-5.4.

1. Объясните концепцию шаблонов в C++ и приведите примеры их использования для создания универсальных классов или функций.
2. В чем разница между статическим и динамическим полиморфизмом?

3. Что такое условные операторы? Приведите пример использования `if` и `else` в коде.
4. Что такое цикл? Объясните разницу между `for` и `while` циклами.
5. Что такое массив? Как его можно использовать в программе?
6. Объясните, что такое функции и каковы их преимущества.
7. Что такое обработка исключений? Приведите пример, когда она может быть полезна.
8. Что такое объектно-ориентированное программирование (ООП)? Назовите основные его принципы.
9. Что такое компилятор и интерпретатор? Каковы их основные отличия?
10. Объясните концепцию инкапсуляции в объектно-ориентированном программировании.
11. Что такое массивы и списки? В чем их основные отличия?
12. Каковы основные принципы работы со строками в программировании?
13. Что такое рекурсия? Приведите пример рекурсивной функции.
14. Что такое библиотека и фреймворк в программировании? Приведите примеры.
15. Что такое база данных? Какие основные типы баз данных существуют?
16. Объясните разницу между статической и динамической типизацией в языках программирования.
17. Что такое тестирование программного обеспечения? Назовите основные его типы
18. Что такое указатель в C++ и как он объявляется?
19. Объясните разницу между `int` и `double`.
20. Что делает оператор `new` и как освобождается память, выделенная с его помощью?
21. В чем разница между `const int x = 5;` и `int const x = 5;`?
22. Что такое перегрузка функций и как она реализуется?
23. Объясните концепцию инкапсуляции в ООП.
24. Что такое наследование в ООП и какие типы наследования существуют?
25. Что такое виртуальные функции и для чего они используются?
26. В чем разница между `std::vector` и `std::array`?
27. Что такое итератор и как он используется для работы с контейнерами STL?
28. Назовите три основных контейнера STL и опишите их назначение.
29. Что такое исключения в C++ и как их обрабатывают?
30. Что такое шаблон функции и как он объявляется?
31. Что такое лямбда-выражение и какой синтаксис у него?
32. Как использовать `std::cout` для вывода данных на консоль?

Критерии оценки, шкала оценивания по контрольным вопросам

Оценка	Описание
«отлично»	Знает весь учебный материал, отлично понимает и прочно усвоил его. На вопросы (в пределах программы) дает правильные, сознательные и уверенные ответы. В устных ответах пользуется литературно правильным языком и не допускает ошибок
«хорошо»	Знает весь требуемый учебный материал, хорошо понимает и прочно усвоил его. На вопросы (в пределах программы) отвечает без затруднений. В устных ответах пользуется литературным языком и не делает грубых ошибок
«удовлетворительно»	Знает основной учебный материал. На вопросы (в пределах программы) отвечает с затруднением. В устных ответах допускает ошибки при изложении материала и в построении речи
«неудовлетворительно»	Не знает большей части учебного материала, отвечает, как правило, лишь на наводящие вопросы преподавателя, неуверенно. В устных ответах допускает частые и грубые ошибки

Критерии оценки, шкала итогового оценивания (зачет)

Оценка	Описание
«зачтено»	Демонстрирует полное понимание проблемы. Знает основные понятия в рамках обсуждаемого вопроса, методы изучения и их взаимосвязь между собой, практические проблемы и имеет представление о перспективных направлениях разработки рассматриваемого вопроса
«не зачтено»	Демонстрирует непонимание проблемы. Не знает основные понятия, методы изучения, в рамках обсуждаемого вопроса не имеет представления об основных практических проблемах

5. Процедура проведения промежуточной аттестации

Промежуточная аттестация по дисциплине проводится в форме зачета. Зачет включает в себя: собеседование по контрольным вопросам.