**ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ**

**(альтернативная дисциплина)**

**1. Актуальность дисциплины и место в программе высшего профессионального образования**

Технологии программирования в настоящее время стремительно развиваются и активно участвуют во всех сферах жизни общества. Компьютеры, мобильные устройства, Интернет, социальные сети, мессенджеры и электронные информационные ресурсы стали неотъемлемой частью не только экономики, но и военных. В этой области приобретение основ программирования военнослужащими, рациональное использование цифровых технологий также сыграет важную роль в увеличении военного потенциала государства.

Предмет «Технологии программирования» включен в блок дисциплин по выбору на 3 курсе.

Чтобы освоить этот предмет, курсанты должны обладать знаниями по программирования, а также русскому, английскому языкам и математике.

Знания, полученные в области технологии программирования, облегчают поиск электронной теоретической информации по всем дисциплинам, служат основой для постоянного совершенствования знаний, быстрого и качественного выполнения поставленных боевых задач.

**2. Цели и задачи дисциплины**

Основная цель преподавания предмета – теоретически и практически объяснить структуру, процесс работы и этапы создания современных средств и программного обеспечения ИКТ при подготовке высококвалифицированных офицеров в области информационных технологий в различных областях.

В задачу предмета «Технологии программирования» входит формирование специализации и профессиональных навыков курсантов и достижение следующих целей:

Обучение обучаемых и обучающихся устройству средств и программного обеспечения ИКТ, программному обеспечению современных устройств, их использованию, а также использованию их современных возможностей;

Творчески приобретать самостоятельные знания, формировать навыки и квалификации, направлять их на эффективное использование объектно-ориентированного программирования и программных средств ИКТ для усиления боевой подготовки и безопасности в войсках.

Формирование практических навыков и компетенций: создание программных разработок военного назначения с использованием ИКТ и компьютерных средств и программно-техническое обеспечение современных средств защиты в различных областях, формирование знаний и умений, связанных с военно-техническими и программными комплексами, современными техническими устройствами, программным обеспечением. инструменты и работа с ними, совершенствование, освоение навыков и компетенций работы с цифровыми устройствами.

В ходе и по окончании освоения предмета, обучающиеся будут владеть следующими навыками:

* структура, функции и основные параметры языков программирования в программной части;
* концепции программного обеспечения и программирование на языке C++ технических средств безопасности;
* классы, структуры и ассоциации;
* управление конфигурацией программного обеспечения;
* по методам тестирования программного обеспечения и обеспечения качества; иметь воображение;
* выбор алгоритмов, подходящих для данной задачи;
* разработка структуры программы;
* устранение неполадок и управление ошибками в программе;
* создание и управление графическим интерфейсом пользователя;
* разработка интерфейса с учётом особенностей заданного объекта;
* знать, как работать с классами, структурами и объединениями и уметь их использовать;
* применение основ современных языков программирования и программных сред;
* применение простых и сложных структур языков программирования;
* оценка алгоритмов, выбор алгоритма решения поставленной задачи, обоснование выбора и применение алгоритма;
* приобрести навыки использования технологии программирования.

**3. Содержание учебной дисциплины**

**Тема 1: Введение и основные понятия «Tехнология программирования».**

Основные понятия. Арифметические операторы в C++. Строковые операторы и методы в С++. Условный оператор в C++ и его применение. Решение задач со строками с использованием условного оператора в C++. Написание программ с использованием условного оператора.

**Тема 2: Работа с операторами более высокого уровня в языке программирования C++.**

Программирование повторяющихся процессов на C++. Программирование повторяющихся процессов на C++. Написание программ с использованием операторов for, while, break и continue на C++. Работа с массивамы и методы работы с ними в C++. Концепция функции в C++. Пользовательская функция.Создание функциональных программ на C++.

**Тема 3: Работа с файлами и каталогами на C++.**

Работа с файлами и каталогами. Открытие файла. Методы работы с файлами. Возможности модуля ОС. Измените путь к файлу и каталогу. Функции и методы работы с каталогами и файлами.Решение задачи с файлами и каталогами.

**Тема 4: Введение и основные понятия ООП в C++.**

Основы ООП в C++. Основы ООП. Объявление и создание экземпляров классов. Класс и объект. Конструктор класса.Методы \_\_init\_\_() и \_\_del\_\_(). Наследование. Специальные методы. Свойства класса.

**Тема 5: Создание графических приложений с использованием C++ Builder.**

Знакомство с C++ Builder и его возможностями. Возможности C++ Builder. Установка C++ Builder. C++ Builder. Виджет QLabel. Виджет QLabel; Шрифт, размер и свойства текста QLabel. C++ Builder. Виджет QLineEdit. Виджет QLineEdit; Метод setStyleSheet(). Создание программы для решения задач различной сложности с использованием C++ Builder. Модальное диалоговое окно C++ Builder. Работает с виджетом MessageBox. Изображения и меню в C++ Builder. Создание дизайна текстового редактора. Написание программы текстового редактора. Написание программы для функций элементов дизайна текстового редактора в C++ Builder. Тестирование программы.

**Тема 6: Введение в сетевое программирование на C++.**

Создание клиент-серверной программы, обменивающейся информацией по сети.Введение в основные методы модуля Socket. Работа с модулем Socket. Знакомство с основными методами модуля Socket; .socket(), .bind, .listen, .accept (), .connect(), .send(), recv(), .close(). Создание клиент-серверной программы TCP на C++. Тестирование программного обеспечения клиент-сервер TCP. Создание современного чат-приложения с использованием C++ Builder. Завершите создание чат-приложения с использованием пакета GUI на C++. Поддержка пакетов, специфичных для приложений C++. Создание загружаемого файла из приложения C++ с помощью пользовательского пакета.

**4. Организационно-методические рекомендации по преподаванию дисциплины.**

«Технологии программирования» используются инновационные педагогические технологии, побуждающие студентов мыслить самостоятельно и свободно, совершенствовать логико-алгоритмическое мышление и навыки говорения, четко и лаконично излагать свою точку зрения по той или иной проблеме, а также интерактивные методы, такие как «Бумеранг», «Пошаговое», «Мозговой штурм», «Карусель», «3 х 4», «Проблема», «Лабиринт», «Блисс-опрос », «Скоробей», «Интерактивная беседа», «Т-схема», «Кластер», «FSMU», «VEN-диаграмма», SWOT -анализ» и другие методы.

Изложение лекционного материала должно быть самостоятельным и целостным, логически связанным с ранее представленными материалами и ориентированным на применение в других дисциплинах и практике. В ходе практических занятий курсанты должны научиться применять полученные теоретические знания.

Каждая лекция включает введение, основную часть и заключение.

Введение включает в себя: название темы, основную идею и значимость темы лекции; цели обучения; учебные вопросы лекции; связь с предыдущими и последующими учебными занятиями; роль лекции в процессе подготовки офицеров на курсах морской пехоты.

Основная часть лекции представляет содержание учебных вопросов. Каждый теоретический аспект лекции должен быть обоснован и доказан наиболее подходящими методами. При изложении основной части лекции обязательным требованием является опора на доказательства, позволяющие студентам объяснить логику развития, закрепления и перехода от абстракции к конкретике научных идей. Содержание основной части каждой лекции должно быть фундаментальным.

В лекции, направленные на практические цели, целесообразно включать практические рекомендации по решению профессиональных и учебных задач.

Каждый учебный вопрос должен заканчиваться кратким изложением теории и практики перспектив развития, логически подводя к следующему учебному вопросу.

В заключительной части лекции подводятся итоги и кратко излагается содержание основной части, указываются области и границы применения теории и практики, а также ставятся вопросы и задания для самостоятельного изучения и обсуждения на последующих семинарских и других видах обучения.

Ведущим методом обучения является устное представление учебного материала с использованием на лекциях кино- и видеофильмов, рисунков, плакатов, макетов, инструментов и моделей.

При выборе темпа подачи материала преподаватель должен учитывать категорию обучающихся (студенты, курсанты), наличие учебной, научной, методической литературы по данной теме (направлению) и другие факторы.

Групповые занятия проводятся с целью дальнейшего закрепления теоретических знаний, полученных на лекциях по данному предмету, формирования навыков самостоятельного использования основных компонентов и операторов языка программирования C++, развития знаний по C++ Builder и Machine Learning, а также формирования базы для обучения курсантов их практическому применению и совершенствованию существующих кодов. Групповые занятия проводятся в специальных аудиториях, в аудиториях с компьютерами, оснащенными необходимым программным обеспечением для языка программирования C++.

Групповые учебные курсы отличаются от других видов обучения использованием широкого спектра учебных средств и учебных пособий для преподавания Технологии программирования, основ языка программирования C++, того, как использовать C++ Builder, как использовать необходимые пакеты для Технологии программирования, а также как их применять, эксплуатировать, обслуживать и чинить.

Используя индивидуальный и коллективный подход, преподаватель находит решения проблемных вопросов, содержащихся в лекции, посредством беседы.

Для активизации изучаемых учебных материалов полезно знакомить учащихся с методическими приемами и проводить обмен идеями между ними, например: «зачем это делается», «насколько удобно (целесообразно, целесообразно)».

практических занятиях курсанты создают программы на современных языках программирования на современных компьютерах и изучают анализ программ.

Практические занятия проводятся в специальных аудиториях, оснащенных современной компьютерной техникой и мультимедийными средствами. Студенты совершенствуют свои умения и навыки посредством теоретического опыта и практических работ.

С целью индивидуализации занятий и повышения качества обучения группы делятся на несколько подгрупп в зависимости от количества инструментов и распределяются по местам обучения.

Практические занятия должны включать элементы соревнования, конкуренции и здорового состязания для вовлечения курсантов в выполнение нормативов.

Учитывая возрастающие требования к ускорению образовательного процесса, необходимо постоянно совершенствовать методику организации и проведения обучения.

В ходе самостоятельной работы курсанты изучают рекомендуемую литературу, пополняют свои конспекты, закрепляют полученные знания.

**5. Самостоятельное обучение и самостоятельная работа.**

|  |  |  |
| --- | --- | --- |
| **Т/р** | **Тема для самостоятельного изучения** | **Форма окончательной работы** |
| 1 | Линейное и разветвленное программирование | Выполняет практически. Готовит отчёт. |
| 2 | Рекурсия и оператор выбора | Выполняет практически. Готовит отчёт. |
| 3 | Указатели и массивы | Выполняет практически. Готовит отчёт. |
| 4 | Работа с файлами. Текстовые файлы | Выполняет практически. Готовит отчёт. |

**6. Основная и дополнительная учебная литература, и источники информации.**

**Основная литература**

1. Джесc Либерти. “Освой самостоятельно С++ за 21 день”;
2. Стефан Р.Девис. С++ для “Чайников”, 5-е издание.: Пер.с англ.- М.: Издательский дом, “Вильямс”, 2007. – 384 с. ISBN 978-5-8459-0723-3 (рус.).

**Рекомендуемые дополнительная литература**

1. Ш.А. Назиров, Р.В. Қобилов “Обектга мўлжалланган дастурлаш”. Ташкент 2013;
2. Introduction to C++ (and C) Programming January 2006;
3. Хаври Дейтел, Пол Дейтел. Как программировать на С++.

**Рекомендуемые** **интернет-сайты**

1. https://ravesli.com/uroki-cpp/
2. https://www.youtube.com/watch?v=ePE0bbsBHv4&list=PLnV3K-pmuXwheCPhhTTBWouWi303FLcz9
3. https://www.w3schools.com/CPP/default.asp