МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное**

**образовательное учреждение высшего образования**

**«Национальный исследовательский Нижегородский государственный университет   
им. Н.И. Лобачевского»**

|  |
| --- |
| Институт информационных технологий, математики и механики |

|  |
| --- |
| УТВЕРЖДАЮ: |

|  |  |  |
| --- | --- | --- |
| Директор |  | В.П. Гергель |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| « |  | » |  |  | 2017 г. |

**Рабочая программа дисциплины**

|  |
| --- |
| **Методы объектно-ориентированного программирования** |

Уровень высшего образования

|  |
| --- |
| **Бакалавриат** |

Направление подготовки

|  |
| --- |
| **01.03.02 Прикладная математика и информатика** |

Направленность образовательной программы

|  |
| --- |
| **Прикладная математика и информатика (общий профиль)** |

Квалификация

|  |
| --- |
| **Бакалавр** |

Форма обучения

|  |
| --- |
| **очная** |

Нижний Новгород

2017

**1. Место и цели дисциплины в структуре ОПОП**

Дисциплина читается во втором семестре бакалавриата (Б1.В.13 – относится к вариативной части блока Б1 ОПОП по направлению подготовки 01.03.02 «Прикладная математика и информатика»). Для изучения дисциплины требуется знания математики и информатики в пределах школьной программы, а также знаний, предусмотренных изучением дисциплины «Языки и методы программирования» во втором семестре бакалавриата.

**Цель освоения дисциплины**

Дисциплина «Методы объектно-ориентированного программирования» является второй частью двухгодичного курса по различным аспектам программирования, **общей целью которого** является подготовка высококвалифицированных разработчиков сложных программных систем моделирования объектов и явлений реального мира, управления экономико-социальными и производственными процессами, а также решения других задач автоматизации, научных исследований и проектирования на основе применения современной вычислительной техники.

Данная дисциплина **преследует цель** систематического изучения следующих аспектов:

* общие вопросы создания программ, включая основные этапы процесса разработки и используемые средства;
* краткие сведения о среде исполнения программ;
* краткие сведения об инструментах программирования;
* основные элементы и принципы построения языков программирования высокого уровня на примере С и С++;
* базовые алгоритмы и основы алгоритмизации с примерами на C;
* различные способы описания моделей объектов предметной области с помощью конструирования типов данных с использованием средств C и C++;
* вопросы динамического управления памятью и работы с файлами с использованием средств C и C++;
* технологии разработки: структурная, модульная, объектно-ориентированная, и их поддержка в языках программирования.

Для достижения целей курса в ходе обучения ставятся следующие **задачи**:

* формирование у студента целостного представления о процессе разработки программы как последовательности взаимно влияющих друг на друга этапов моделирования, проектирования и реализации, начиная с постановки задачи до обеспечения развития программы в процессе жизненного цикла;
* освоение приёмов разработки типовых алгоритмов (вычислительного и нечисленного характера), современного стиля программирования, методов отладки программ;
* обеспечение глубокого понимания целей и тенденций развития технологий программирования и их основных концепций;
* изучение технологий модульного и структурного программирования, вопросов проектирования данных, а также **основ объектно-ориентированного программирования**;
* формирование у студентов адекватного представления о современных требованиях к программному продукту, особенно в части реализации интерфейса с пользователем и изучение методов проектирования и реализации интерфейса;
* освоение современных средств проектирования и реализации программ, современных языков программирования на примере C, C++ и соответствующих сред программирования (Microsoft Visual Studio).

1. **Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)**

|  |  |
| --- | --- |
| **Формируемые компетенции** | **Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций** |
| *ПК-7- Способность к разработке и применению алгоритмических и программных решений в области*  *системного и прикладного программного обеспечения*  *(Начальный этап)* | ***ЗНАТЬ***  *З1 (ПК-7) общие принципы использования вычислительной техники для решения прикладных задач;*  *З2 (ПК-7) основные идеи, лежащие в основе современных языков программирования и основные составляющие современного языка программирования на примере C и C++;*  *З3 (ПК-7) технологии разработки программного обеспечения (структурное и модульное программирование – подробно, объектно-ориентированное программирование – на уровне общих идей) и способы их выражения в языках программирования (C, ООП – в C++);*  *З4 (ПК-7) методы конструирования новых типов данных (C, C++).*  *З5 (ПК-7) методы и приемы работы с динамической памятью (C, C++);*  *З6 (ПК-7) методы работы с внешней памятью (C, C++).*  ***УМЕТЬ***  *У1(ПК-7) разрабатывать и отлаживать программы средней сложности с использованием языков программирования C, C++ и технологий структурного, модульного и объектно-ориентированного программирования.*  ***ВЛАДЕТЬ***  *В1(ПК-7) современными интегрированными средами разработки программ;*  *В2 (ПК-7) навыками реализации, тестирования и отладки программных систем средней сложности (на базовом уровне).* |

1. **Структура и содержание дисциплины «Методы объектно-ориентированного программирования»**

Объем дисциплины составляет 6 зачетных единиц, всего 216 часов, из которых 128часов составляет контактная работа обучающегося с преподавателем:

64 часа занятий лекционного типа

32 часа занятий семинарского типа

32 часа занятий лабораторного типа

Самостоятельная работа обучающегося 88 часов (в т.ч. 54 часа подготовки к экзамену)

Содержание дисциплины:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Наименование и краткое содержание разделов и тем дисциплины,**  **форма промежуточной аттестации по дисциплине** | **Всего**  **(часы)** | В том числе | | | | |
| **Контактная работа (работа во взаимодействии с преподавателем), часы**  из них | | | | **Самостоятельная**  **работа обучающегося, часы** |
| **Занятия**  **лекционного типа** | **Занятия**  **семинарского типа (практические занятия)** | **Занятия**  **лабораторного типа** | **Всего** |  |
| Введение в предмет. История языка C, C и C++, обзор литературы. | **10** | 4 | 1 | 1 | **6** | **4** |
| Элементы языка C, от Pascal к C шаг за шагом. Переменные, типы данных, константы, литералы, перечисления, объявления, операции, математические функции, преобразование типов, ввод/вывод. | **16** | 8 | 2 | 2 | **12** | **4** |
| Структурное программирование и операторы языка C. | **11** | 2 | 2 | 2 | **6** | **5** |
| Массивы и указатели в C. Работа со строками. | **15** | 6 | 2 | 2 | **10** | **5** |
| Модульное программирование в C. | **13** | 4 | 2 | 2 | **8** | **5** |
| Структуры и объединения в C. | **9** | 2 | 1 | 1 | **4** | **5** |
| Работа с файлами в C. | **13** | 4 | 2 | 2 | **8** | **5** |
| Отличия стандартов C89 и С99 | **7** | 2 | – | – | **2** | **5** |
| Язык программирования С++. Отличия от C. Литература | **13** | 4 | 2 | 2 | **8** | **5** |
| Концепция объектного подхода. Основные идеи объектного подхода на примерах. | **11** | 2 | 2 | 2 | **6** | **5** |
| Понятие класса. Классы в C++. Поля и методы. Разграничение доступа. Классы и объекты. Константные методы. | **13** | 4 | 2 | 2 | **8** | **5** |
| Конструкторы. | **13** | 4 | 2 | 2 | **8** | **5** |
| Деструкторы. | **11** | 2 | 2 | 2 | **6** | **5** |
| Статические поля и методы. Дружественные функции. | **11** | 2 | 2 | 2 | **6** | **5** |
| Перегрузка операций. | **13** | 4 | 2 | 2 | **8** | **5** |
| Наследование и агрегация | **12** | 3 | 2 | 2 | **7** | **5** |
| Виртуальные методы. Абстрактные классы. | **12** | 3 | 2 | 2 | **7** | **5** |
| Введение в шаблоны | **13** | 4 | 2 | 2 | **8** | **5** |
| **В т.ч. текущий контроль** | **2** |  |  |  |  |  |
| **Промежуточная аттестация: Экзамен** | | | | | | |

**Тематика практических занятий**

**Программирование на языке C.**

1. *Структура программы на языке С*. Структура программы. Функция языка С, функция main. Объявления переменных. Типы данных. Операции языка С. Ввод/вывод (С). Блок в языке С. Время жизни и область видимости переменных.
2. *Структурное программирование*. Условный оператор. Условие в языке С. Переключатель. Цикл с предусловием, постусловием, цикл for.
3. *Массивы и указатели*. Синтаксис объявления. Массив как параметр функции. Указатели, операции с указателями. Указатели на массивы.
4. *Динамическое распределение памяти*. Выделение, освобождение памяти. Динамические массивы. Указатели на указатели.
5. *Работа со строками*. Представление строк, базовые алгоритмы.
6. *Многофайловая программа. Алгоритмы сортировки*. Директивы #include, #define, #ifndef. Указатель на функцию. Создание статической библиотеки.
7. *Структуры.* Структура как абстрактный тип данных. Структуры как параметр и возвращаемое значение функции.
8. *Дополнительные темы:* Основы системного программирования. Битовые операции. Макросы.

**Элементы объектно-ориентированного программирования.**

1. *Некоторые отличия С++ от С*. Ввод/вывод, константы, работа с динамической памятью, ссылки, передача параметров по ссылке, перегрузка функций, константные параметры.
2. *Классы и объекты*. Объявление класса, поля и методы, спецификаторы доступа. Описание класса. Конструктор по умолчанию и конструктор-инициализатор. Понятие деструктора. Объявление и использование объектов. Указатель this.
3. *Перегрузка операций*. Введение. Перегрузка операций, два способа перегрузки операций. Правило перегрузки арифметических операций. Динамические поля и перегрузка операций.
4. *Сложные случаи при создании и удалении объектов*. Использование полей-указателей и полей-объектов в классах, выделение и освобождение памяти. Конструктор копирования. Перегрузка операций присваивания, сравнения, индексирования, ввода/вывода в поток.
5. *Перегрузка операций и конструктор преобразования типа*.
6. *Агрегация и наследование*. Агрегация, наследование и спецификаторы доступа. Создание и удаление объектов. Перегрузка и перекрытие методов.
7. *Виртуальные методы*. Виртуальное перекрытие методов. Абстрактные методы и классы.
8. *Шаблоны*. Шаблоны функций. Введение в шаблоны классов
9. *Введение в STL*.
10. *Обработка исключений*.
11. *Дополнительные темы*: Статические и динамические библиотеки. Интеграция программ на С и С++. Приведение типов в С++. Пространства имен.
12. **Образовательные технологии**

**Обучение данной дисциплине** организовано следующим образом:

1. Предусмотрены 3 вида занятий: лекционные занятия, занятия семинарского типа (практические занятия) и лабораторные занятия.

2. В рамках лекционных занятий основное внимание уделяется формированию у студентов целостного понимания истории развития, текущего состояния и перспективных направлений в области базового программирования, развитию кругозора. Лекционное время преимущественно расходуется не на разбор синтаксиса конкретных конструкций языков программирования, а на изучение основных принципов и концепций, лежащих в основе языков и технологий программирования, прагматике использования тех или иных алгоритмических и технических решений (в пределах рамок курса).

3. В рамках практических занятий основное внимание уделяется изучению синтаксиса языков, их основных элементов, со студентами обсуждаются возможные способы решения типовых учебных задач, проводится их сравнительный анализ, прототипируются наиболее важные решения, ставятся задачи для самостоятельной работы. В рамках практики основное внимание уделяются развитию у студентов навыков программирования и отладки, обсуждаются типичные ошибки, методы их обнаружения и устранения, проверяются программы студентов, изучается текущий прогресс в обучении.

4. Самостоятельная работа студентов в ходе всего учебного года предполагает выполнение ряда практических работ (проектов). При этом в каждой практической работе студенты проходят весь путь, начиная от постановки учебной задачи до сдачи преподавателю работающей программы с краткой документацией. Сдача практических работ и консультирование студентов по поводу различных аспектов их выполнения проходит в рамках занятий в компьютерном классе.

1. **Учебно-методическое обеспечение самостоятельной работы обучающихся**
2. **Виды самостоятельной работы студентов**

Самостоятельная работа студентов подразделяется на следующие категории:

– Изучение учебной литературы (см. перечень образовательных материалов).

– Решение учебных задач (основы алгоритмизации, закрепление знаний синтаксиса и семантики языков программирования, развитие навыков практического программирования).

– Выполнение практических работ на следующие темы (по усмотрению преподавателя):

* Практическая работа 1. Решение систем алгебраических линейных уравнений.
* Практическая работа 2. Разработка класса «Комплексное число».
* Практическая работа 3. Разработка класса «Вектор».

– Студенты, демонстрирующие хорошую успеваемость и высокий уровень мотивации, могут принять участие в конкурсе программных проектов, ежегодно проводимом кафедрой Математического обеспечения и суперкомпьютерных технологий института ИТММ.

1. **Образовательные материалы для самостоятельной работы студентов**
2. Калинина Н., Костюкова Н. Курс «Основы программирования на языке C». <http://www.intuit.ru/studies/courses/43/43/info>
3. Фридман А. Курс «Язык программирования C++». <http://www.intuit.ru/studies/courses/17/17/info>
4. Баженова И., Сухомлин В. Курс «Введение в программирование». <http://www.intuit.ru/studies/courses/27/27/info>
5. Страуструп Б. Курс «Язык программирования C++ для профессионалов». <http://www.intuit.ru/studies/courses/98/98/info>
6. **Фонд оценочных средств для промежуточной аттестации по дисциплине**, включающий:
   1. Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

*Оценка уровня формирования компетенции (ПК-7)*

|  |  |  |
| --- | --- | --- |
| **Индикаторы компетенции** | **Критерии оценивания (дескрипторы)** | **Шкала оценивания** |
| ***ЗНАТЬ***  *З1 (ПК-7) общие принципы использования вычислительной техники для решения прикладных задач;*  *З2 (ПК-7) основные идеи, лежащие в основе современных языков программирования и основные составляющие современного языка программирования на примере C и C++;*  *З3 (ПК-7) технологии разработки программного обеспечения (структурное и модульное программирование – подробно, объектно-ориентированное программирование – на уровне общих идей) и способы их выражения в языках программирования (C, ООП – в C++);*  *З4 (ПК-7) методы конструирования новых типов данных (C, C++).*  *З5 (ПК-7) методы и приемы работы с динамической памятью (C, C++);*  *З6 (ПК-7) методы работы с внешней памятью (C, C++).*  ***УМЕТЬ***  *У1(ПК-7) разрабатывать и отлаживать программы средней сложности с использованием языков программирования C, C++ и технологий структурного, модульного и объектно-ориентированного программирования.*  ***ВЛАДЕТЬ***  *В1(ПК-7) современными интегрированными средами разработки программ;*  *В2 (ПК-7) навыками реализации, тестирования и отладки программных систем средней сложности (на базовом уровне).* | Отсутствие знаний материала, отсутствует способность решения стандартных задач, полное отсутствие навыков, предусмотренных компетенцией. | Плохой уровень формирования компетенции.  «Плохо» (семестр 2) |
| Наличие грубых ошибок в основном материале, наличие грубых ошибок при решении стандартных задач, отсутствие навыков, предусмотренных данной компетенцией. | Неудовлетворительный уровень формирования компетенции.  «неудовлетворительно» (семестр 2) |
| **Знать** некоторые основные понятия, изучаемые в рамках дисциплины (З1–З7).  **Уметь** (У1) с погрешностями. **Владеть** некоторыми основными навыками, демонстрируя их в стандартных ситуациях. | Удовлетворительный уровень формирования компетенции.  «Удовлетворительно»  (семестр 2) |
| **Знать** большинство основныхпонятий,изучаемых в рамках дисциплины (З1–З7).  **Уметь** (У1) с незначительными погрешностями.  **Владеть** основными навыками, демонстрируя их в стандартных ситуациях. | Хороший уровень  формирования компетенции.  «Хорошо» (семестр 2) |
| **Знать** основные понятия, изучаемые в рамках дисциплины (З1–З7).  **Уметь** (У1) с незначительными погрешностями.  **Владеть** всеми основными навыками, демонстрируя их в стандартных ситуациях, в том числе при решении дополнительных задач. | Очень хороший уровень  формирования компетенции  «Очень хорошо» (семестр 2) |
| **Знать** основной материал, предусмотренный компетенцией, без ошибок и погрешностей. **Уметь** (У1) в полном объеме. **Владеть** всеми навыками, демонстрируя их не только в стандартных ситуациях, но и при решении нестандартных задач | Отличный уровень  формирования компетенции  «Отлично» (семестр 2) |
| **Знать** основной и дополнительный материал, предусмотренный компетенцией, без ошибок и погрешностей. **Уметь** (У1) в полном объеме. **Владеть** всеми навыками, демонстрируя их не только в стандартных ситуациях, но и при решении нестандартных задач. | Превосходный уровень  формирования компетенции  «Превосходно»  (семестр 2) |

**Карта компетенций для оценивания умений и навыков**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Индикаторы  компетенции | Критерии оценивания (дескрипторы) | | | | | | | |
| «плохо» | «неудовлетворительно» | «удовлетворительно» | «хорошо» | «очень хорошо» | | «отлично» | «превосходно» |
| Умения  У1(ПК-7) | Отсутствует способность решения стандартных задач | Наличие грубых ошибок при решении стандартных задач | Способность решения основных стандартных задач с негрубыми ошибками | Способность решения всех стандартных задач с незначительными погрешностями | Способность решения всех стандартных задач без ошибок и погрешностей | Способность решения стандартных и некоторых нестандартных задач | | Способность решения стандартных задач и широкого круга нестандартных задач |
| Навыки  В1(ПК-7), В2(ПК-7) | Полное отсутствие навыков, предусмотренных компетенцией | Отсутствие ряда важнейших навыков, предусмотренных данной компетенцией | Наличие минимально необходимого множества навыков | Наличие большинства основных навыков, продемонстрированное в стандартных ситуациях | Наличие всех основных навыков, продемонстрированных в стандартных ситуациях | Наличие всех навыков, продемонстрированное в стандартных ситуациях | | Наличие всех навыков, продемонстрированное в стандартных и нестандартных ситуациях |
| Личностные качества | Соответствующие личностные качества не сформированы | Сформированность личностных качеств недостаточна для достижения основных целей обучения | Сформированность личностных качеств минимально необходимая для достижения основных целей обучения | Личностные качества в целом сформирваны | Сформированные личностные качества достаточны для достижения целей обучения | Личностные качества сформированы на высоком уровне | | Сформированность личностных качеств выше обязательных требований |

* 1. Описание шкал оценивания

Для оценивания результатов учебной деятельности студентов при изучении дисциплины используется следующая комбинированная система оценивания, цель которой состоит в следующем:

– дополнительно мотивировать студентов изучать необходимый материал;

– объективно оценивать знания студентов.

Результаты работы студентов оцениваются непрерывно в ходе семестра. При этом учитываются следующие факторы:

1. *Посещаемость студентами практических занятий*.
2. *Результаты текущей успеваемости*. Студенты должны сдать все предусмотренные практические работы (проекты). Общее число и содержание таких работ устанавливается преподавателем практики по согласованию с лектором. Применяется следующая схема: преподаватель практики выставляет студенту оценку по практике (ОП) на основании работы в семестре.

Оценка результатов на экзамене выполняется следующим образом:

1. В случае если ОП не является удовлетворительной, студент начинает сдачу экзамена с решения задачи. Иначе, студент имеет выбор из двух альтернатив:
   * + Согласиться с оценкой по практике; в этом случае студент начинает сдачу экзамена с ответа на теоретические вопросы.
     + Попробовать повысить оценку по практике; в этом случае студент начинает сдачу экзамена с решения задачи, в результате чего данная оценка может быть как повышена, так и понижена (в зависимости от результатов).
2. Если итоговая ОП не хуже, чем «удовлетворительно», экзамен продолжается. При этом студент берет билет, готовится к ответу и отвечает по билету, состоящему из двух вопросов. В результате он получает оценку ОБ по теории. Далее оценки ОП и ОБ усредняются. В случае выставления оценок «отлично» и «превосходно» в соответствии с критериями, изложенными в пункте 6.1, студенту предлагается решить дополнительную задачу. Также должны быть заданы дополнительные вопросы на понимание материала.

Таким образом, сформированность компетенции ПК-7 в части «уметь» и «владеть» проверяется по результатам работы в семестре или при решении задачи на экзамене. Сформированность компетенции ПК-7 в части «знать» проверяется по результатам собеседования на экзамене по заранее подготовленным билетам.

|  |  |
| --- | --- |
| **Оценка** | **Уровень подготовки** |
| Превосходно | На экзамене обучаемый показал высокий уровень подготовки, безупречное владение теоретическим материалом, знания, выходящие за рамки рассмотренного в курсе материала, творческий поход к разрешению нестандартных ситуаций. Имеет полностью и творчески выполненный практикум в семестре. Демонстрирует способность решать дополнительные предложенные задачи, требующие оригинальности мышления.  Освоение материала на 100%. |
| Отлично | Обучаемый показал высокий уровень владения материалом курса. Имеет выполненный лабораторный практикум. Демонстрирует способность решать дополнительные предложенные задачи, требующие оригинальности мышления.  Освоение материала на уровне 90-99%. |
| Очень хорошо | В целом – весьма хорошая подготовка. Обучаемый дает ответы на все теоретические вопросы билета, но с рядом ошибок и неточностей; может решать задачи из всех основных разделов, имеет выполненный практикум.  Освоение материала на уровне 80-90%. |
| Хорошо | Достаточно хорошая подготовка, но с заметными ошибками или недочетами; получен полный ответ на все теоретические вопросы билета, но с рядом ошибок. Практические задания обучаемый выполняет, но с недочетами, практикум, в основном, выполнен.  Освоение материала на уровне 60-79%. |
| Удовлетворительно | Минимально достаточный уровень подготовки. Обучаемый в значительной части отвечает на все вопросы билета, но с множеством ошибок, не носящих грубого характера; имеет задолженности или низкую оценку по практикуму; предложенные практические задания выполняет с заметными ошибками.  Освоение материала на уровне 35-59%. |
| Неудовлетворительно | Подготовка не достаточна и требует дополнительного изучения материала. Студент дает ошибочные ответы, как на теоретические вопросы билета, так и на наводящие и дополнительные вопросы экзаменатора. Как правило, имеет задолженности по практикуму и не может правильно решить предложенные практические задачи.  Освоение материала порядка 15-34%. |
| Плохо | Подготовка абсолютно недостаточна. Обучаемый не отвечает на поставленные вопросы, не понимает терминологию; имеет задолженности по лабораторному практикуму, не знает подходов к решению практических задач.  Освоение материала ниже 15%. |

* 1. Критерии и процедуры оценивания результатов обучения по дисциплине (модулю), характеризующих этапы формирования компетенций

Для оценивания результатов обучения в виде знаний используются устные ответы на вопросы.

Для оценивания результатов обучения в виде умений и владений используются следующие процедуры и технологии:

–выполнение студентами практических работ, включающих постановку одной достаточно сложной учебной задачи в виде краткой формулировки действий, которые следует выполнить, и описания результата, который нужно получить.

– решение студентами экзаменационных задач.

* 1. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции.

*Примеры контрольных заданий для проведения текущего контроля успеваемости, а также для выставления оценки по практике на экзамене (ОП) для оценивания результатов формирования ПК-7*

1. Реализовать функцию на языке C, которая решает следующую задачу: дан массив целых чисел. Сформировать из него другой массив, содержащий элементы первого ровно в одном экземпляре (массив без повторений).
2. Реализовать функцию на языке C, которая решает следующую задачу: дана строка. Вернуть данную строку с обратным порядком символов. Вспомогательных строк не создавать.
3. Записать объявление класса комплексное число на языке C++, перегрузить операцию умножения.

*Примеры задач для оценки компетенции «ПК-7» в части «уметь», «владеть» на экзамен*

*Задача 1*. Задан массив A, содержащий n вещественных чисел. Проверить, является ли массив упорядоченным по возрастанию.

*Задача 2*. Задан массив A, содержащий n целых чисел и вещественное число R. Найти индексы двух элементов массива, сумма которых наиболее близка к числу R.

*Пример тестовых вопросов в качестве допуска к экзамену для оценки компетенции ПК-7*

**1. Тип вопроса:** одиночный выбор

Укажите правильный вариант объявления переменной целого типа в языках C и C++:

* 1. int a; (+)
  2. float a;
  3. double a;

1. **Тип вопроса:** одиночный выбор

Укажите, что будет выведено на экран в результате работы программы на языке C (приведен фрагмент функции main()):

int a;

int \*pa;

a = 1;

pa = &a;

printf(“%d”, \*pa);

* 1. 0
  2. 1 (+)
  3. -1
  4. Ошибка времени исполнения

1. **Тип вопроса:** одиночный выбор

Укажите значение переменной S в результате работы программы на языке C (предполагается, что все переменные объявлены, приведен фрагмент функции main()):

s = 0;

for (i = 0; i < 5; i++)

s = s + i;

* 1. 0
  2. 2
  3. 5
  4. 10 (+)

*Примеры экзаменационных билетов по курсу для оценивания результатов формирования ПК-7*

Национальный исследовательский Нижегородский государственный университет

им. Н.И. Лобачевского

Институт ИТММ

Кафедра математического обеспечения и суперкомпьютерных технологий

Дисциплина «Методы объектно-ориентированного программирования»

**ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 21**

1. Особенности использования массивов в качестве параметров процедур и функций.
2. Перегрузка унарных и бинарных операций. Перегрузка операций + и += для класса Complex. Примеры использования.
3. Задача.

Зав. кафедрой Р.Г. Стронгин

Экзаменатор И.Б. Мееров

Национальный исследовательский Нижегородский государственный университет

им. Н.И. Лобачевского

Институт ИТММ

Кафедра математического обеспечения и суперкомпьютерных технологий

Дисциплина «Методы объектно-ориентированного программирования»

**ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 22**

1. Тип указатель на функцию. Примеры использования.
2. Перегрузка унарных и бинарных операций. Перегрузка оператора присваивания на примере класса Vector. Отличия оператора присваивания от конструктора копирования. Примеры использования.
3. Задача.

Зав. кафедрой Р.Г. Стронгин

Экзаменатор И.Б. Мееров

* 1. Методические материалы, определяющие процедуры оценивания

Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся в ННГУ от 13.02.2014. <http://www.unn.ru/site/images/docs/obrazov-org/Formi_stroki_kontrolya_13.02.2014.pdf>

Положение о фонде оценочных средств, утвержденное приказом ректора ННГУ от 10.06.2015 №247-ОД.

1. **Учебно-методическое и информационное обеспечение дисциплины**

а) основная литература:

1. Страуструп Б. Курс «Язык программирования C++ для профессионалов». <http://www.intuit.ru/studies/courses/98/98/info>

б) дополнительная литература:

1. Калинина Н., Костюкова Н. Курс «Основы программирования на языке C». <http://www.intuit.ru/studies/courses/43/43/info>
2. Фридман А. Курс «Язык программирования C++». <http://www.intuit.ru/studies/courses/17/17/info>
3. **Материально-техническое обеспечение дисциплины**

Имеются в наличии учебные аудитории для проведения занятий лекционного типа (оснащенные проектором), занятий семинарского типа, промежуточной аттестации, а также помещения для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет». Для лабораторного практикума имеется терминал-класс.

Наличие рекомендованной литературы.

Используемое лицензионные программное обеспечение:

* Операционные системы семейства Microsoft Windows, лицензия по подписке Microsoft Imagine.
* Среда разработки семейства Microsoft Visual Studio, лицензия по подписке Microsoft Imagine.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.02 «Прикладная математика и информатика»

Авторы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_И.Б. Мееров,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Сысоев

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Заведующий кафедрой МОСТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Р.Г. Стронгин

Программа одобрена на заседании методической комиссии Института информационных технологий, математики и механики ННГУ им. Н.И. Лобачевского

от 29 августа 2017 года, протокол № 20