**Осенний конкурс образовательных инноваций 2020**

**Номинация: Оригинальные элементы online-преподавания**

**Автор: Горденко Мария Константиновна, старший преподаватель, младший научный сотрудник, аспирант, Факультет компьютерных наук**

**Общая идея и цель проведения курса с использованием представленной методики**

В данной разработке описаны методы и инструменты проведения лекционных и семинарских занятий в режиме **онлайн** факультатива «Введение в программирование». Данный курс является адаптационным и предназначен для студентов 1 курса бакалавриата образовательной программы «Программная инженерия». В ходе дисциплины студенты знакомятся с языком программирования C#, учатся писать программы, в том числе и в небольших группах, изучают необходимый теоретический материал.

Для реализации курса в онлайн формате используются следующие техники и методики:

* Перевернутый класс;
* Групповые дискуссии;
* Командные упражнения;
* Олимпиада;
* Традиционные формы проведения лекций и семинаров.

Для реализации курса в онлайн формате используется следующее программное обеспечение:

* Kahoot;
* Socrative;
* PeerGrade;
* GitHub Classroom;
* MS Teams;
* Repl.it;
* Telegram;
* NearPod/PearDeck.

**Организация занятий**

В рамках курса предусмотрено проведение как лекционных, так и семинарских занятий. Все занятия проводятся в системе MS Teams[[1]](#footnote-1). Все описанные здесь инструменты рассматриваются подробнее в расширенном описании заявки.

Условно, проводимые в рамках данного курса лекции можно классифицировать как:

* Традиционные лекции;
* Лекции-дискуссии;
* Лекции по модели «перевернутый класс»;
* Видео-лекции.

Если говорить про семинарские занятия, то организация их в онлайн-формате требует определенного рода изобретательности, т.к. необходимо вовлечь студентов в образовательный процесс, а из-за того что они находятся дома, а не в аудитории, то сделать это гораздо сложнее. В рамках данного курса проводятся следующие активности на семинарских занятиях:

* групповые дискуссии;
* обсуждение кейсовых ситуаций в командах;
* командные упражнения;
* соревновательные элементы на дополнительный балл (олимпиады);
* совместное решение тестовых вопросов с последующим обсуждением;
* совместное решение задач с последующим обсуждением;
* самостоятельное решение студентами тестов и задач.

Более подробное описание каждой лекции и семинарской активности приведено в расширенном описании заявки.

**Порядок оценивания работы студентов, элементы контроля**

В ходе курса предусмотрено три основных вида элементов контроля и экзамен:

* тест;
* лабораторная работа (со взаимным оцениванием в PeerGrade);
* лабораторная работа (автоматическая оценка в системе GitHub Classroom).

Рассмотрим подробнее каждый элемент контроля.

**Тест** – это набор вопросов с одним или множественным выбором, а также вопросы со словесным ответом. Тесты проводятся в системе Socrative, что позволяет получить автоматически оценку, без ручной проверки преподавателем. В ходе курса предусмотрено 2 теста на оценку (**Т1** и **Т2**).

**Лабораторная работа (со взаимным оцениванием в PeerGrade, Л1 и Л3)** представляет из себя задачу, которая дается студентам на одну неделю. Студенты пишут код и загружают обратно в систему. После чего, студентам дается три дня на проверку работ своих сокурсников. Каждый студент проверяет по пять работ сокурсников и выставляет оценки в соответствии с заранее заданными критериями. Работа студента получает средний балл из всех выставленных отметок. Стоит отметить, что т.к. это ребята 1 года обучения, то процесс оценивания необходимо корректировать, в каких-то ситуациях, работы необходимо пересматривать. Однако, как показывает практика, если заранее обсудить все критерии, то проблем с оценками не возникает.

**Лабораторная работа (автоматическая оценка в системе GitHub Classroom, Л2)** представляет из себя набор из 5-7 задач на одну неделю. Студенты решают данные задачи и загружают в систему GitHub Classroom. До окончания периода сдач студентам доступен ограниченный набор тестовых ситуаций, на которых проверяются программы. После окончания периода сдач задачи проверяются на всех тестах и студенту выставляется балл в соответствии с баллами, набранными на тестах задач.

Экзамен состоит из двух частей – решение теста в Socrative и лабораторная работа из 3 задач с автоматической оценкой в системе GitHub Classroom.

Также предусмотрены неоцениваемые авто проверяемые домашние задания в системе GitHub Classroom. Автоматическая проверка позволяет студентам получить мгновенную обратную связь и самостоятельно найти ошибку либо же задать вопрос преподавателю в чате MS Teams или Telegram-чате, а не ждать, когда преподаватель проверит работу.

По всем видам работ выставляется десятибалльная оценка.

Оценка за лабораторную работу выставляется как средняя результатов взаимного оценивания. Одиночный результат взаимного оценивания формируется согласно оценочному листу для каждой лабораторной работы.

Итоговая оценка (**ИО**) по дисциплине «Введение в программирование» вычисляется по формуле:

**ИО = 0,7\*Н + 0,3\*ЭК**, где **Н**– оценка, накопленная за лабораторные работы первого и второго модулей, ЭК – оценка за итоговую (экзаменационную) контрольную работу.

**Н = 0,2\*Л1 + 0,2\*Т1 + 0,2\*Л2+ 0,2\*Л3 + 0,2\*Т2**

В случае, если студент получил дополнительный балл за участие в игре-олимпиаде, то он прибавляется к оценке **ИО**. Стоит отметить, что оценка **ИО** не может быть более 10 баллов.

*Округление оценок при вычислениях осуществляется до ближайшего целого.*

**План курса и организации занятий**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название раздела | Тип лекции | Работа на семинаре | Домашнее задание и элемент контроля |
|
| 1 | Алгоритмизация и программирование на языке C# | Традиционная лекция | 1) Совместное решение задач с последующим обсуждением  2) Совместное решение тестовых вопросов с последующим обсуждением | ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |
| 2 | Основные алгоритмические конструкции | Традиционная лекция | 1) Командное упражнение  2) Групповая дискуссия  3) Самостоятельное решение студентами задач | Контроль – Т1  ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |
| 3 | Особенности решения задач с циклами | Лекция-дискуссия | 1) Обсуждение кейсовых ситуаций в команде  2) Самостоятельное решение студентами задач  3) Совместное решение тестовых вопросов с последующим обсуждением | ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |
| 4 | Структурная декомпозиция | Традиционная лекция | 1) Самостоятельное решение студентами задач и тестов  2) Командная работа | Контроль – Л1 |
| 5 | Рекуррентные соотношения. Рекурсия. | Видео-лекция | Соревновательный элемент на дополнительный балл (олимпиада по мотивам игры «Математическая карусель») | ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |
| 6 | Одномерные массивы | Лекция по модели «перевернутый класс» | 1) Командное упражнение  2) Групповая дискуссия  3) Совместное решение задач с последующим обсуждением | Контроль – Л2.  ДЗ: посмотреть видео-лекцию, прочитать учебник. |
| 7 | Многомерные массивы | Видео-лекция | 1) Групповая дискуссия  2) Самостоятельное решение студентами задач  3) Совместное решение тестовых вопросов с последующим обсуждением | Контроль – Т2  ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |
| 8 | Принципы разработки. Основы объектно-ориентированного программирования | Лекция-дискуссия | 1) Самостоятельное решение студентами задач  2) Совместное решение тестовых вопросов с последующим обсуждением | ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |
| 9 | Отношения между классами | Традиционная лекция | Совместное решение тестовых вопросов и задач с последующим обсуждением | Контроль – Л3.  ДЗ: посмотреть видео-лекцию, прочитать учебник. |
| 10 | Абстрактные типы данных. Структуры данных | Лекция по модели «перевернутый класс» | Соревновательный элемент на дополнительный балл (олимпиада по мотивам игры «Математическая карусель») | ДЗ: 1) автоматически проверяемый набор задач в GitHub Classroom;  2) чтение учебника |

**Способы взаимодействия со студентами и организация обратной связи**

Основным каналом взаимодействия со студентами является команда в MS Teams.

Среди достоинств данной системы можно отметить её универсальность, т.к. она позволяет проводить занятия, хранить материалы, работать с календарем и планировщиком задач, а самое главное, что интегрирована с корпоративной почтой студентов в домене @edu.hse.ru и не требует дополнительных регистраций, а также студентам и преподавателям предоставляется бесплатно.

Кроме того, со студентами был создан Telegram-канал для оперативного решения вопросов. Однако, со студентами была договоренность решать все вопросы посредством чатов в MS Teams, но бывают ситуации, когда вопрос нужно решить очень срочно и студенты пользуются чатом Telegram.

Традиционным каналом взаимодействия является почта и преподаватель отвечает на вопросы по почте, однако, в основном, студенты задают все вопросы в MS Teams.

С учетом того, что в онлайн-формате обратная связь может потерять в силу того, что преподаватель не видит, а зачастую и не слышит студентов, то в ходе данного курса сбор обратной связи происходит либо путем разнообразных опросов (например, в Telegram или MS Teams) по желаемым темам, темпу обучения и т.д, либо в последние 5 минут занятия обязательно проводится рефлексия и собирается обратная связь студентов исходя из которой могут перерабатываться последующие занятия. Также студенты знают, что преподаватель открыт к диалогу и если стесняются озвучить свои проблемы, то пишут их в личные сообщения в MS Teams.

**Полученные результаты**

На текущий момент, с сентября 2020 года при таком разнообразном формате проведения занятий с большой долей вовлеченности студентов удалось несмотря на то, что курс является факультативным и посещаемость никак не учитывается при расчете оценки, сохранить посещаемость на уровне 92% (как лекционных, так и семинарских занятий). Помимо посещаемости стабильно высокий процент обучающихся, выполняющих домашние неоцениваемые задания. Скорее всего, этому причиной мгновенный фидбек. Разнообразие форм занятий и применяемых инструментов позволяет вовлечь студентов и «разговорить» их на занятии, что даже в офлайне было сделать довольно трудно.

**Масштабирование на другие предметы**

Описанные методики не являются новыми и уже успешно применяются в рамках разных курсов. Однако изюминкой данной разработки является их совместное применение в рамках одного курса для поддержки образовательно процесса в онлайн, а также подобранные инструменты для их реализации.

Все приведенные методики могут быть масштабированы и на другие курсы за счет своей гибкости.

1. <https://teams.microsoft.com> [↑](#footnote-ref-1)