#### Задание 1

****Цель задания****

Познакомиться с принципами работы цикла for и закрепить их на практике.

****Что нужно сделать****

1. Создайте новый проект и назовите его FactorialCalculator.
2. Создайте класс Main и метод main.
3. Создайте переменную типа int и получите её значение из консоли:

int value = new Scanner(System.in).nextInt();

1. Напишите программу, которая будет считать факториал переданного в консоли числа. Факториал целого числа — это произведение этого целого числа на все целые положительные числа меньше него. Обозначается восклицательным знаком. Вот примеры расчётов факториала для некоторых чисел:  
   3! = 1 × 2 × 3 = 6  
   5! = 1 × 2 × 3 × 4 × 5 = 120
2. Для проверки кода используйте числа от 1 до 12.

****Критерии оценки****

Зачёт: программа работает, как показано в примере выше, и в ней используется цикл for.

На доработку: задание не выполнено.

****Как отправить задание на проверку****

Пришлите ZIP-архив проекта с выполненным заданием. Для этого воспользуйтесь командой экспорта в самой среде разработки. Найдите её в меню: ****File → Export → Project to ZIP File****… и затем выберите папку, куда сохранить. Полученный архив отправьте преподавателю.

#### Задание 2

****Цель задания****

Познакомиться с принципами работы цикла for и закрепить их на практике.

****Что нужно сделать****

1. Создайте новый проект и назовите его ForCycle.
2. Создайте класс Main и метод main.
3. Создайте переменную типа int и получите её значение из консоли:

int value = new Scanner(System.in).nextInt();

1. Напишите программу, которая будет находить все пары положительных целых чисел, произведение которых будет равно значению value. Например, если ввести число 6, программа должна выдать следующий результат:  
   1\*6  
   2\*3  
   3\*2  
   6\*1

****Рекомендации по выполнению****

Для написания программы используйте два цикла for — один, вложенный в другой.

****Критерии оценки****

Зачёт: программа работает, как показано в примере выше, и в ней используются циклы for.

На доработку: задание не выполнено.

****Как отправить задание на проверку****

Пришлите ZIP-архив проекта с выполненным заданием. Для этого воспользуйтесь командой экспорта в самой среде разработки. Найдите её в меню: ****File → Export → Project to ZIP File****… и затем выберите папку, куда сохранить. Полученный архив отправьте преподавателю.

#### Задание 3

****Цель задания****

Познакомиться с принципами работы цикла while и закрепить их на практике.

****Что нужно сделать****

1. Создайте новый проект и назовите его SwimmingPool.
2. Создайте переменную int volume, в которой будет храниться объём бассейна, и установите её равной 1200.
3. Представьте, что система наполняет бассейн со скоростью 30 литров в минуту, но у бассейна не отключён слив, и через него каждую минуту выливается 10 литров. Создайте ещё две переменные, в которых будут храниться эти значения:

int fillingSpeed = 30; //30 litres per minute  
  
int devastationSpeed = 10; //10 litres per minute

1. Напишите программу с использованием цикла while, которая рассчитает и выведет в консоль время наполнения бассейна с учётом скоростей наполнения и опустошения.

****Рекомендации по выполнению****

Создайте переменную, в которой будет храниться текущий объём бассейна, и каждую итерацию цикла добавляйте к нему fillingSpeed и вычитайте из него devastationSpeed. Как только текущий объём бассейна станет равен значению переменной volume, выполните команду break. В процессе выполнения цикла считайте количество итераций — это и будет время наполнения бассейна.

****Критерии оценки****

Зачёт: программа работает, как показано в примере выше, и в ней используется цикл while.

На доработку: задание не выполнено.

****Как отправить задание на проверку****

Пришлите ZIP-архив проекта с выполненным заданием. Для этого воспользуйтесь командой экспорта в самой среде разработки. Найдите её в меню: ****File → Export → Project to ZIP File****… и затем выберите папку, куда сохранить. Полученный архив отправьте преподавателю.

#### Задание 4

****Цель задания****

Познакомиться с принципами работы цикла while и закрепить их на практике.

****Что нужно сделать****

1. Создайте новый проект GuessTheNumber. Напишите в нём программу-игру, в которой пользователь будет угадывать некое число, а программа, если он не угадал, будет сообщать ему, больше или меньше введённое им число, чем загаданное.
2. В начале программы создайте переменную int с именем value, в которой будет храниться угадываемое число.
3. Напишите бесконечный цикл с использованием оператора while, в котором вы будете получать введённое пользователем число из консоли.

int attempt = new Scanner(System.in).nextInt();

1. Напишите проверку этого числа: если число равно загаданному, выводите в консоль сообщение «Вы угадали!» и завершайте цикл.
2. Если пользователь не угадал число, выводите в консоль подсказку: больше или меньше введённое число, чем загаданное.
3. Пример работы программы:

****Рекомендации по выполнению****

Если хотите создать более реальную программу, нужно, чтобы загадываемое число генерировалось случайным образом в диапазоне от 0 до 100:

int value = new Random().nextInt(100);

****Критерии оценки****

Зачёт: программа работает, как показано в примере выше, и в ней используется цикл while.

На доработку: задание не выполнено.

****Как отправить задание на проверку****

Пришлите ZIP-архив проекта с выполненным заданием. Для этого воспользуйтесь командой экспорта в самой среде разработки. Найдите её в меню: ****File → Export → Project to ZIP File****… и затем выберите папку, куда сохранить. Полученный архив отправьте преподавателю.