ПРИЛОЖЕНИЕ А
ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ «Математическая логика и теория алгоритмов»

1. Перечень оценочных средств для компетенций, формируемых в результате освоения дисциплины

Код контролируемой компетенции	Способ оценивания	Оценочное средство
ПК-2: Способность оценивать временную и емкостную сложность программного обеспечения	Зачет	Комплект контролирующих материалов для зачета

2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Оцениваемые компетенции представлены в разделе «Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций» рабочей программы дисциплины «Математическая логика и теория алгоритмов».

При оценивании сформированности компетенций по дисциплине «Математическая логика и теория алгоритмов» используется 100-балльная шкала.

Критерий	Оценка по 100-балльной шкале	Оценка по традиционной шкале
Студент освоил изучаемый материал, выполняет задания в соответствии с индикаторами достижения компетенций, может допускать отдельные ошибки.	25-100	*Зачтено*
Студент не освоил основное содержание изученного материала, задания в соответствии с индикаторами достижения компетенций не выполнены или выполнены неверно.	0-24	*Не зачтено*

3. Типовые контрольные задания или иные материалы, необходимые для оценки уровня достижения компетенций в соответствии с индикаторами

1.Задача на реализацию оператора примитивной рекурсии (задача 1).

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.1 Выбирает методы оценки временной и емкостной сложности программного обеспечения
	ПК-2.2 Выбирает или конструирует алгоритмы для решения прикладных задач с учетом оценки их временной и емкостной сложности

Напишите рекурсивную и нерекурсивную программы для вычисления значения функции f, полученной оператором примитивной рекурсии над функциями $g(x) = x \dot{-} 1$ и $h(x, y, t) = x \cdot t + y$. Вычислите значение функции f в точке $(0, 3)$. Учитывая оценки временной и емкостной сложности, поясните, какой алгоритм вычисления функции f будет более эффективным.

2.Задача на реализацию оператора примитивной рекурсии (задача 2).

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.1 Выбирает методы оценки временной и емкостной сложности программного обеспечения
	ПК-2.2 Выбирает или конструирует алгоритмы для решения прикладных задач с учетом оценки их временной и емкостной сложности

Напишите рекурсивную и нерекурсивную программы для вычисления значения функции f, полученной оператором примитивной рекурсии над функциями $g(x) = x + 2$ и $h(x, y, t) = (t \dot{-} 1) \cdot (x + y)$. Вычислите значение функции f в точке $(3, 4)$. Учитывая оценки временной и емкостной сложности, поясните, какой алгоритм вычисления функции f будет более эффективным.

3.Задача на определение порядка временной сложности (задача 1).

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.1 Выбирает методы оценки временной и емкостной сложности программного обеспечения

Программа на языке Си содержит следующую функцию

```
float F(int N)
{
   float x=0;
   if (N<4) return 1;
   for (int i=0; i<N; i++)
      for (int j=0; j<N; j++) x += sin(N*i-j) * cos(N*j+i);
   return x;
}
```

Определите порядок временной сложности вычисления этой функции по приведенной выше программе.

4.Задача на определение порядка временной сложности (задача 2).

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.1 Выбирает методы оценки временной и емкостной сложности программного обеспечения

Программа на языке Си содержит следующую функцию

```
int F(int N)
{
if (N<4) return 1;
   else return F(N-1) + F(N-3) + 1;
}
```

Определите порядок временной сложности вычисления этой функции по приведенной выше программе.

5.Задача на определение порядка временной сложности (задача 3).

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.1 Выбирает методы оценки временной и емкостной сложности программного обеспечения

Программа на языке Си содержит следующую функцию

```
float F(int N)
{
if (N<3) return 1;
   else return F(N-1) * sin(N-3);
}
```

Определите порядок временной сложности вычисления этой функции по приведенной выше программе.

6.Задача на построение машины Тьюринга.

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.2 Выбирает или конструирует алгоритмы для решения прикладных задач с учетом оценки их временной и емкостной сложности

Постройте машину Тьюринга, вычисляющую функцию $f(x, y) = x + 2$. Выберите алгоритм для решения задачи с учетом оценки временной и емкостной сложности. В качестве меры временной сложности используйте количество выполненных команд машины Тьюринга для перехода из начальной конфигурации в заключительную. Емкостную сложность вычислений на машине Тьюринга определите количеством ячеек ленты, которые заполнены непустыми символами либо посещались головкой во время работы.

7.Задача на разработку алгоритма Маркова.

Компетенция	Индикатор достижения компетенции
ПК-2 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-2.2 Выбирает или конструирует алгоритмы для решения прикладных задач с учетом оценки их временной и емкостной сложности

Разработайте алгоритм Маркова, вычисляющий функцию $f(x, y) = \frac{2x + y}{3}$. Выберите алгоритм для решения задачи с учетом оценки временной и емкостной сложности. В терминах алгоритмов Маркова временная сложность соответствует числу правил, которые были применены, а емкостная – максимальной длине слова, которое возникает в процессе выполнения алгоритма.

4. Файл и/или БТЗ с полным комплектом оценочных материалов прилагается.