

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ЛПУ»)**

**Структурное подразделение** Институт физико-математического образова-  
ния, информационных и обслуживающих технологий  
**Кафедра** информационных образовательных технологий и систем

**УТВЕРЖДАЮ**

Врио директора ИФМОИОТ

 Е.А. Журавлёва

« 15 »  20 25 г.


Приложение к рабочей программе учебной дисциплины

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
для проведения текущего контроля и промежуточной аттестации  
обучающихся по дисциплине  
**Практикум по решению олимпиадных задач по информатике**

По направлению подготовки 44.03.05 Педагогическое образование (с двумя  
профилями подготовки)

Профиль подготовки Физика. Информатика  
Квалификация выпускника бакалавр  
Форма обучения очная  
Курс 5

Разработчик  
ассистент Тивоненко А.А.

заведующий кафедрой информацион-  
ных образовательных технологий и систем  
 Д.А. Капустин

Протокол

от « 14 »  20 25 г. № 5

Луганск, 2025

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

## 1.1. Область применения

Фонд оценочных средств (ФОС) – неотъемлемая часть рабочей программы дисциплины «Практикум по решению олимпиадных задач по информатике» и предназначен для контроля и оценки образовательных достижений студентов, освоивших программу дисциплины.

## 1.2. Цели и задачи фонда оценочных средств

Цель ФОС – установить соответствие уровня подготовки обучающегося требованиям ФГОС ВО – бакалавриат по направлению подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки), утвержденным приказом Министерства образования и науки Российской Федерации от 22.02.2018 г. № 125 (с изменениями и дополнениями) и Профессиональным стандартом, утвержденным Приказом Министерства труда и социальной защиты Российской Федерации «Об утверждении профессионального стандарта “Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)”» от 18 октября 2013 г. № 544н (с изменениями и дополнениями); «Об утверждении профессионального стандарта “Педагог дополнительного образования детей и взрослых”» от 22 сентября 2021 г. № 652н., соответствующих профессиональной деятельности выпускников.

## 1.3. Перечень компетенций, формируемых в процессе освоения основной образовательной программы

Процесс освоения дисциплины направлен на формирование следующих компетенций и индикаторов их достижения:

Код по ФГОС ВО	Индикатор достижения
Универсальные	
Общепрофессиональные	
Профессиональные	
ПК-1. Способен осваивать и использовать теоретические знания и практические умения и навыки в предметной области при решении профессиональных задач	ПК-1.1. Знает структуру, состав и дидактические единицы предметной области (преподаваемого предмета). ПК-1.2. Умеет осуществлять отбор учебного содержания для его реализации в различных формах обучения в соответствии с требованиями ФГОС ВО. ПК-1.3. Демонстрирует умение разрабатывать различные формы учебных занятий, применять методы, приемы и технологии обучения, в том числе информационные.

#### 1.4. Этапы формирования компетенций и средства оценивания уровня их сформированности

Этапы формирования компетенций	Компетенции	Контрольно-оценочные средства / способ оценивания
Тема 1. Организация олимпиад по информатике	ПК-1	Устный опрос.
Тема 2. Методика подготовки к олимпиадам по информатике	ПК-1	Устный опрос.
Тема 3. Особенности подготовки задач для олимпиад по информатике	ПК-1	Устный опрос.
Тема 4. Обзор типовых алгоритмов решения олимпиадных задач	ПК-1	Выполнение лабораторных работ
Текущая аттестация	ПК-1	Индивидуальное задание
Промежуточная аттестация	ПК-1	Зачет

#### 1.5. Описание показателей формирования компетенций

Код компетенции	Результаты сформированности
	Знает: Умеет: Владеет навыками:
	Знает: Умеет: Владеет навыками:
ПК-1	Знает: синтаксис языка программирования – основные понятия и конструкции ООП, принципы написания программы с использованием ООП. Умеет: использовать прикладное программное обеспечение общего и целевого назначения, в частности VisualStudio 2022 и выше, умеет производить отладку программ. Владеет навыками: методами составления, описания и реализации алгоритмов; принципами и методами написания программ с использованием ООП; принципами отладки программ.

#### 1.6. Критерии оценивания компетенций на разных этапах их формирования

Вид учебной работы	Количество баллов		
	ОФО	О-ЗФО	ЗФО
Устные ответы на семинарских занятиях	20		
Выполнение и защита практических / лабораторных работ	25		
Самостоятельная работа	10		
Иные виды учебной работы (подготовка презентации, написание реферата, решение задач и др.)	15		
Зачет	30		

<b>Всего</b>	<b>100</b>
--------------	------------

### Накопительная система оценивания по 100-балльной шкале

Четырехбалльная система оценивания экзамена	100-балльная шкала	Буквенная шкала, соответствующая 100-балльной шкале	Система оценивания зачета
Отлично	<b>90-100</b>	<b>А</b> – отлично – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом сформированы; все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	Зачтено
Хорошо	<b>83-89</b>	<b>В</b> – очень хорошо – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом в основном сформированы; все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному	
Хорошо	<b>75-82</b>	<b>С</b> – хорошо – теоретическое содержание курса освоено полностью; некоторые практические навыки работы с освоенным материалом сформированы недостаточно; все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	
Удовлетворительно	<b>63-74</b>	<b>Д</b> – удовлетворительно – теоретическое содержание курса освоено частично, но пробелы не носят существенного характера; необходимые практические навыки работы с освоенным материалом в основном сформированы; большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки	
Удовлетворительно	<b>50-62</b>	<b>Е</b> – посредственно – теоретическое содержание курса освоено частично; некоторые практические навыки работы не сформированы, многие предусмотренные учебной программой обучения учебные задания не выполнены либо качество выполненных некоторых из них оценено числом баллов, близким к минимальному	
Неудовлетворительно	<b>21-49</b>	<b>FX</b> – неудовлетворительно – теоретическое содержание курса освоено частично; необходимые практические навыки работы с освоенным материалом не сформированы; большинство предусмотренных учебной программой обучения учебных заданий не выполнено либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительно самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	Не зачтено
Неудовлетворительно	<b>0-20</b>	<b>F</b> – неудовлетворительно – теоретическое	

		<p>содержание курса не освоено;  необходимые практические навыки работы  не сформированы; все выполненные  учебные задания содержат грубые ошибки;  дополнительная самостоятельная работа  над материалом курса не приведет к  какому-либо значимому повышению  качества выполнения учебных заданий</p>	
--	--	---	--

## **2. КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА**

### **2.1. Оценочные средства текущего контроля**

1. Виды алгоритмов, свойства алгоритмов, способы описания алгоритмов. Оценка сложности алгоритмов.
2. Нормативно-правовая база организации олимпиад по информатике.
3. Системы автоматизированного проведения турниров.
4. Основные методы, используемые для подготовки учащихся к олимпиадам.
5. Классификация олимпиадных задач по информатике.
6. Особенности олимпиадных задач. Требования к системе задач школьной олимпиады.
7. Подготовительный этап олимпиады. Проведение олимпиады. Проверка заданий, способы оценки. Подведение итогов.
8. Работа с системами автоматизированного проведения турниров по информатике
9. Типовые алгоритмы решения олимпиадных задач (задачи для начинающих)
10. Алгоритмы на дату и время для решения олимпиадных задач
11. Алгоритмы со строками решения олимпиадных задач
12. Задачи на формулы
13. Переборные алгоритмы
14. Алгоритмы на поиск
15. Сортировки в олимпиадных задачах
16. Жадные алгоритмы
17. комбинаторные алгоритмы
18. Задачи на бинарный поиск по ответу
19. Динамическое программирование
20. Задачи на поиск в ширину и поиск в глубину
21. Геометрические задачи в олимпиадном программировании

### **2.2. Оценочные средства для промежуточной аттестации**

1. Виды алгоритмов
2. Свойства алгоритмов, способы описания алгоритмов.
3. Оценка сложности алгоритмов.
4. Анализ задачи и выбор подхода.
5. Разбиение задачи на подзадачи.
6. Классические и олимпиадные задачи.
7. Нормативно-правовая база организации олимпиад по информатике.
8. Системы автоматизированного проведения турниров.
9. Основные методы, используемые для подготовки учащихся к олимпиадам.
10. Классификация олимпиадных задач по информатике.
11. Особенности олимпиадных задач.
12. Требования к системе задач школьной олимпиады.
13. Подготовительный этап олимпиады.
14. Проведение олимпиады.
15. Проверка заданий, способы оценки. Подведение итогов.
16. Работа с системами автоматизированного проведения турниров по информатике
17. Типовые алгоритмы решения олимпиадных задач (задачи для начинающих)
18. Алгоритмы на дату и время для решения олимпиадных задач
19. Алгоритмы со строками решения олимпиадных задач
20. Структуры данных в задачах
21. Задачи на формулы
22. Переборные алгоритмы
23. Алгоритмы на поиск

24. Алгоритмы на указатели
25. Задачи на бинарный поиск
26. Сортировки в олимпиадных задачах
27. Жадные алгоритмы
28. Комбинаторные алгоритмы
29. Задачи на бинарный поиск по ответу
30. Динамическое программирование
31. Геометрические задачи в олимпиадном программировании
32. Решение задач с использованием динамического программирования
33. Что такое одномерный массив в C++ и как его объявить?
34. Как объявить и инициализировать одномерный массив с элементами типа `int` в C++?
35. Как получить доступ к элементу массива в C++?
36. Как объявить и инициализировать двумерный массив (матрицу) в C++?
37. Как перебрать все элементы двумерного массива с помощью вложенных циклов в C++?
38. Что такое структура (структура данных) в C++ и как она объявляется?
39. Как объявить переменную типа структуры и получить доступ к ее членам в C++?
40. Что такое перечисление (`enum`) в C++ и как оно объявляется?
41. Как использовать перечисление в C++?
42. Что такое объединение (`union`) в C++ и как оно объявляется?
43. Как использовать объединение в C++?
44. Как объявить и инициализировать строку (массив символов) в C++?
45. Зачем нужна библиотека `string`?
46. Как получить длину строки в C++?
47. Как сконкатенировать две строки в C++?
48. Как скопировать одну строку в другую в C++?
49. Как проверить, содержит ли строка определенный подстроку в C++?
50. Как найти первое вхождение символа или подстроки в строке в C++?
51. Как осуществить поиск всех вхождений символа или подстроки в строке в C++?
52. Что такое динамический массив (`dynamic array`) в C++ и как он создается?
53. Как добавить элемент в динамический массив в C++?
54. Как удалить элемент из динамического массива в C++?
55. Как изменить размер динамического массива в C++?
56. Что такое вектор (`vector`) в C++ и как он отличается от обычного массива?
57. Как объявить и инициализировать вектор в C++?
58. Как добавить элемент в конец вектора в C++?
59. Как получить доступ к элементам вектора в C++?
60. Как удалить элемент из вектора в C++?
61. Что такое множество (`set`) в C++ и как оно работает?
62. Как объявить и инициализировать множество в C++?
63. Как добавить элемент в множество в C++?
64. Как удалить элемент из множества в C++?
65. Что такое ассоциативный массив (`map`) в C++ и как он работает?
66. Как объявить и инициализировать ассоциативный массив в C++?
67. Как добавить элемент в ассоциативный массив в C++?
68. Как получить доступ к элементам ассоциативного массива в C++?
69. Что такое рекурсия и как она используется в программировании?
70. Рекурсия что это и зачем?
71. Одномерные динамические массивы
72. Двумерные динамические массивы
73. Зачем нужна библиотека `list` в C++ и как она работает?
74. Как объявить и инициализировать список с помощью `list` в C++?

75. Как добавить элемент в список с помощью list в C++?
76. Как удалить элемент из списка с помощью list в C++?
77. Зачем нужна библиотека stack в C++ и как она работает?
78. Как объявить и инициализировать стек с помощью stack в C++?
79. Как добавить элемент в стек с помощью stack в C++?
80. Как удалить элемент из стека с помощью stack в C++?
81. Зачем нужна библиотека queue в C++ и как она работает?
82. Как объявить и инициализировать очередь с помощью queue в C++?
83. Как добавить элемент в очередь с помощью queue в C++?
84. Как удалить элемент из очереди с помощью queue в C++?
85. Зачем нужна библиотека deque в C++ и как она работает?
86. Как объявить и инициализировать дек с помощью deque в C++?
87. Как добавить элемент в дек с помощью deque в C++?
88. Как удалить элемент из дека с помощью deque в C++?
89. Графы ориентированные и неориентированные.
90. Графы ориентированные. Поиск короткого пути – алгоритм Дейкстры
91. Графы ориентированные. Поиск короткого пути – алгоритм Флойда
92. Нахождения минимального остовного дерева. Алгоритм Прима.
93. Поиск в глубину назначение и применение
94. Поиск в ширину назначение и применение
95. Нахождения минимального остовного дерева. Алгоритм Крускала
96. Каким образом можно работать с файлами в C++?
97. Зачем нужна библиотека fstream?
98. Как записывать данные в файл?
99. Как считывать данные из файла?
100. Как открыть и закрыть файл?