

Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Факультет инновационных технологий

УТВЕРЖДЕНО:  
Декан  
С. В. Шидловский

Оценочные материалы по дисциплине

**Программирование (основы) 1**

по направлению подготовки

**09.03.02 Информационные системы и технологии**

Направленность (профиль) подготовки:

**Программное и аппаратное обеспечение беспилотных авиационных систем**

Форма обучения  
**Очная**

Квалификация  
**Инженер - программист**  
**Инженер - разработчик**

Год приема  
**2024**

СОГЛАСОВАНО:  
Руководитель ОП  
С.В. Шидловский

Председатель УМК  
О.В. Вусович

Томск – 2024

## **1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами**

Целью освоения дисциплины является формирование следующих компетенций:

БК-1 Способен применять общие и специализированные компьютерные программы при решении задач профессиональной деятельности

ОПК-5 Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем

ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий

ОПК-7 Способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

РОБК-1.1 Знает правила и принципы применения общих и специализированных компьютерных программ для решения задач профессиональной деятельности;

РОБК-1.2 Умеет применять современные IT-технологии для сбора, анализа и представления информации; использовать в профессиональной деятельности общие и специализированные компьютерные программы;

РООПК-5.1 Знает основы системного администрирования различных систем их обновления и поддержки;

РООПК-5.2 Умеет осуществлять инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем;

РООПК-6.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий;

РООПК-6.2 Умеет разрабатывать алгоритмы и программы, пригодные для практического применения в задачах профессиональной деятельности;

РООПК-7.1 Знает особенности платформ и инструментальных программно-аппаратных средств для реализации информационных систем;

РООПК-7.2 Умеет выбирать платформы и инструментальные программно-аппаратные средства для реализации информационных систем.

## **2. Оценочные материалы текущего контроля и критерии оценивания**

Элементы текущего контроля:

– контрольная работа;

Типовые задания для проведения текущего контроля успеваемости по дисциплине «Программирование (основы) 1»

Типовое задание на Контрольную по 1-му модулю предмета

Разработчик Вася очень не любит длинные названия переменных. Для Васи длинное название - это такое, длина которого состоит больше, чем из десяти символов.

Вася придумал для себя алгоритм замены таких длинных названий на более короткие записывается первая и последняя буква слова, а между ними — количество букв между первой и последней буквой.

Таким образом, переменная “synchrophasotron” преобразуется в переменную “s14n”.

Позже Вася обнаружил недостаток своего алгоритма - существуют переменные сокращенные названия которых совпадают - он назвал это явление колизей. Для

переменных с коллизией он придумал следующий способ решения: в конце сокращенного названия переменной добавляется номер коллизии для неё.

Так, для переменные “synchrophasotron” и “synchrocyclotron” будут выглядеть следующим образом: “s14n” и “s14n1”

Ваша задача написать алгоритм, который заменить все большие названия переменных на сокращенные.

*Формат входных данных*

В первой строке содержится целое число  $n$  ( $1 \leq n \leq 100$ ).

В каждой из последующих  $n$  строк содержится по одному слову.

*Все слова состоят из малых латинских букв и имеют длину от 1 до 100 символов.*

*Формат выходных данных*

Выведите  $n$  строк. В  $i$  строке должен находиться результат замены  $i$ -го слова из входных данных.

Входные данные	Выходные данные
3 code synchrophasotron synchrocyclotron	code s14n s14n1
1 hedgehog	hedgehog
4 word localization internationalization pneumonoultramicroscopicsilicovolcanoconiosis	word l10n i18n p43s

#### Типовое задание на Контрольную по 2-му модулю предмета

Студент ХИТСа в качестве дипломной работы решил взять очень трудоемкий алгоритм. Суммарно у него есть  $n$  дней, чтобы запустить особую программу и представить полученные результаты. Но есть одна проблема: программе нужно работать  $x$  дней, чтобы подсчитать результаты. Но, к счастью, программу можно оптимизировать. Если потратить  $y$  (целое неотрицательное) дней на оптимизацию, то программа станет работать за  $\frac{x}{y+1}$  дней. Программу можно оптимизировать сколько угодно дней, но запустить можно только один раз (т.е. нельзя оптимизировать программу первый раз, потом запустить, потом оптимизировать ещё раз. Можно только вначале оптимизировать, а потом запустить). В целом остаётся главный вопрос: успеет ли студент Хитса представить результаты программы за  $n$  дней.

*Входные данные*

В первой строке задано единственное целое число  $T$  ( $1 \leq T \leq 50$ ) — количество наборов входных данных. В следующих  $T$  строках заданы сами наборы – по одному в строке. Каждая строка содержит два целых числа  $n$  и  $x$  (оба до  $10^9$ ) — количество дней перед дедлайном и количество дней, за которое срабатывает программа.

*Выходные данные*

Выведите T ответов, YES если студент успеет, EPIC FAIL если нет

3	YES
1 1	YES
4 5	EPIC FAIL
5 11	

Типовое задание на Контрольную по 3-му модулю предмета

Вам дано двоичное дерево (слева потомки меньше узла, справа больше узла), в котором все элементы уникальны по значению.

Ваша задача - придумать, как вывести путь от узла со значением X до корня дерева.

На вход даётся значение N - количество узлов дерева. Далее в каждой из N строк для вершины с порядковым номером  $i$  вводится два числа:  $vi$  и  $pi$  - значение вершины и порядковый номер родителя вершины (у корня дерева номер родителя равен нулю).

Затем вводится число K - количество вершин, от которых нужно будет построить путь до корня дерева. В каждой из K последующих строк содержится число m - значение узла. Гарантируется, что узел со значением m присутствует в дереве.

В ответ выведите путь (цепочку порядковых номеров вершин) от каждой из K вершин до корня дерева. Каждый путь должен отделяться новой строкой.

*При решении задачи обязательно необходимо использовать структуру (-ы)*

**Пример**

Входные данные	Выходные данные
9	6 3 1
64 0	9 5 3 1
29 1	1
87 1	
24 2	
68 3	
100 3	
27 4	
65 5	
76 5	
3	
100	
76	
64	

Типовое задание на Контрольную по 4-му модулю предмета

На одном из предметов в Выдающейся Информативной Топовой Школе студентам необходимо сдавать задачи. Всего задач 9, но за раз можно сдать лишь 4 различные задачи. Конечно же, студенты последнего набора понимают важность предмета и ни за что не списывают. И всё же им хочется сформировать очередь на сдачу таким образом, чтобы максимально похожие решения были сданы в разном порядке.

Поэтому они договорились, что проводится сдача будет согласно следующим правилам:

1. Можно увеличить номер первой сдаваемой задачи на 1, если он не равен 9, и если новый номер задачи не встречается в последовательности сдаваемых студентом задач. Если встречается, то номер можно увеличить до ближайшего числа, которое не входит в последовательность.

2. Можно уменьшить номер последней сдаваемой задачи на 1, если он не равен 1 и если новый номер задачи не присутствует в последовательности сдаваемых студентом задач. Если встречается, то номер можно уменьшить до ближайшего числа, которое не входит в последовательность.

3. Можно циклически сдвинуть все номера на один вправо.

4. Можно циклически сдвинуть все номера на один влево.

Например, применяя эти правила к порядку задач 1235, можно получить числа 4235, 1236, 5123 и 2351 соответственно.

Точные правила сдачи студенты пока не придумали, но пока их интересует вопрос, как получить из одного порядка сдачи задач другой за минимальное количество операций.

*Формат входного файла*

Во входном файле содержится два различных четырехзначных числа, каждое из которых не содержит нулей

*Формат выходного файла*

Программа должна вывести последовательность четырехзначных чисел, не содержащих нулей. Последовательность должна начинаться первым из данных чисел и заканчиваться вторым из данных чисел, каждое последующее число в последовательности должно быть получено из предыдущего числа применением одного из правил. Количество чисел в последовательности должно быть минимально возможным.

Input	Output
9876	9876
5987	9875
	5987

### **3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания**

Типовые задания для проведения промежуточной аттестации:

Вам даны  $n$  пар прямоугольников. Каждая пара прямоугольников находится на отдельной плоскости и взаимодействия между парами не происходит...

Вам необходимо для каждой пары подсчитать площадь пересечения прямоугольников и отсортировать все пары в порядке этой площади

*Формат входных данных*

$n$

далее  $n$  строк по 8 чисел, 2 координаты левого верхнего угла первого прямоугольник, 2 координаты правого нижнего угла первого прямоугольника, 2 координаты левого верхнего угла второго прямоугольник, 2 координаты правого нижнего угла второго прямоугольника

*Формат выходных данных*

Порядковые номера пар прямоугольников в отсортированном виде

В волшебном мире есть волшебники, которые умеют делать волшебные сумки разной вместимости, которые умеют вмещать любой объем предметов в них, с самым главным условием, чтобы суммарно эти предметы весили не более  $S$  килограмм(но

каждую сумку можно зачаровать на отдельный вес и нельзя помещать одну такую сумку в другую).

Но поскольку никто не знает как волшебники это делают, то волшебники жадно установили на такие сумки ужасные цены, если сумка может вмещать не более  $S$  килограммов, то цена за такую сумку будет  $S^{\log_{10} S}$  золотых монет... Помимо этого, поскольку волшебникам было лень поддерживать чары на сумки разных размеров правильно, то они добавили предупреждение в руководство пользователя: если предмет весит меньше чем  $\log_{10} S$ , то он растворяется в сумке(и предмета больше не существует)

Эпический герой решил купить для себя такую сумку, но у него возник вопрос... Очевидно, что чем больше он возьмет сумку, тем дороже она ему обойдется, поэтому он решил каждому своему предмету(которые еще и весят по разному) подсчитать полезность. И наш эпический герой будет считать, что покупка сумки была выгодна только если максимальная сумма полезности всех предметов, которые есть у героя и которые он сможет в неё положить(без уничтожения), больше чем стоимость этой сумки. Далее герой решил найти максимальный размер выгодной сумки, но не смог... Теперь ваша задача найти максимально большую выгодную сумку

*Формат входных данных*

$n$  - количество предметов, которые есть у героя, далее для каждого предмета пара чисел полезность и вес этого предмета

*Формат выходных данных*

Искомый размер максимально выгодной сумки

5	10
1 1	
2 2	
3 3	
4 4	
5 5	

P.S. Гарантируется что сумма полезности всех предметов не превысит 100 000 000

Суммарный вес предметов ограничен  $10^{18}$

Ваша задача написать алгоритм для поиска в глубину, который в качестве основного представления графа используя список рёбер

Прим.

1) Вам нельзя преобразовывать граф в матрицу смежности или список смежности. Т.Е. можно использовать или список рёбер или его производные или альтернативное представление списка рёбер

2) Суммарная эффективность алгоритма должна быть примерно такой же как и поиск в глубину выполненный на матрице смежности (допускается небольшое/ухудшение производительности), хотя при эффективной реализации будет быстрее

**4. Информация о разработчиках**

Змеев Денис Олегович, кандидат технических наук, доцент НОЦ «Высшая ИТ школа».