

Московский авиационный институт (национальный исследовательский университет)  
 Вступительный междисциплинарный экзамен в магистратуру  
 по направлению подготовки  
 01.04.02 «Прикладная математика и информатика»  
 Магистерские программы  
 «Математическое и программное обеспечение вычислительных систем» (а)  
 «Компьютерное моделирование и оптимизация информационных систем» (б)  
 Билет 33

1	а)	Задача на программирование. Язык C++. Напишите многопоточную программу, используя std::thread. Программа должна запускать 10 потоков. Каждый из потоков должен выполнять лямбда-функцию, печатающую его номер (от 1 до 10).				
	б)	<p>Определитель. Методы нахождения и свойства.</p> <p><b>Задача.</b> Найдите величину <math>x</math> из уравнения:</p> $\begin{vmatrix} -3 & 9 & 3 & 6 \\ -5 & 8 & 2 & 7 \\ 4 & -5 & -3 & -2 \\ x & -8 & -4 & -5 \end{vmatrix} = 18.$				
2	а)	Задача на программирование. Язык C, C++ или Java. Составить программу удаления узла из двоичного дерева.				
	б)	<p>Определенный интеграл Римана. Формула Ньютона-Лейбница. Методы интегрирования.</p> <p><b>Задача.</b> Вычислите интеграл</p> $\int_2^3 \frac{x^2 + 1}{x^3 - x}.$				
3	а)	<p>Задача на программирование. Языки C, C++ или Java. Составить программу решения задачи. Рассматривая шифровки, полученные от русской радистки Кэт и профессора Плейшнера, Мюллер вдруг вспомнил, как он с двумя стариками, Рафке и Хилли, распутывал в двадцатые годы одно дело. Там тоже была весьма сложная шифровка, которая пересылалась в двух различных экземплярах. Содержимое каждого экземпляра по отдельности не представляло интереса, однако вместе эти шифровки обозначали конкретную информацию о канале, через который связывались заговорщики НСДАП. Чтобы получить ее, нужно было выделить в обеих шифровках некоторую одинаковую последовательность (которая могла разделяться любым количеством любых знаков, как внутри одной части шифровки, так и внутри другой) и вычислить ее длину. При этом заговорщики были настолько хитры, что считали только максимальную длину всех возможных последовательностей, совпадающих в шифровках. Тогда Мюллер разгадал шифр. Сейчас годы брали свое — он видел два листа, в которых, очевидно, было зашифровано одно и то же сообщение, но расшифровать его не мог.</p> <p>Вашей задачей будет не разгадать хитроумный код полковника Исаева, а попытаться расшифровать код заговорщиков НСДАП, так легко раскрытый Мюллером.</p> <p><i>Входные данные</i> В двух строках даны A и B соответственно первая и вторая шифровки, перехваченные у НСДАП. Обе шифровки состоят из латинских букв в верхнем регистре.</p> <p><i>Выходные данные</i> В единственной строке вывода должна содержаться максимальная длина совпадающей последовательности знаков в двух шифровках.</p> <table><tr><td>Пример входных данных</td><td>Пример выходных данных</td></tr><tr><td>ABCBDAB BDCABA</td><td>4</td></tr></table> <p>Ожидаемая временная сложность алгоритма: <math>O( A  B )</math>, где <math> S </math> — длина строки S.</p>	Пример входных данных	Пример выходных данных	ABCBDAB BDCABA	4
Пример входных данных	Пример выходных данных					
ABCBDAB BDCABA	4					

	<p>б) Предельные теоремы теории вероятностей: закон больших чисел, центральная предельная теорема, теорема Муавра-Лапласа.</p> <p><b>Задача.</b> Независимые случайные величины <math>\xi_1, \xi_2, \dots, \xi_{144}</math> одинаково распределены по закону <math>R(-8; -2)</math>. Оцените вероятность <math>P(\xi_1 + \xi_2 + \dots + \xi_{144} &gt; -800)</math>.</p>						
4	<p>а) Задача на программирование. Языки C, C++ или Java. Составить программу решения задачи. Никита решил создать новую операционную систему, он знает, что в любой нормальной операционной системе имеется менеджер пакетов, который позволяет легко устанавливать пакеты, осуществляя управление зависимостями. Одной из задач менеджера пакетов в операционной системе будет контроль за отсутствием некорректных зависимостей между пакетами. Каждый пакет в системе может иметь произвольное множество пакетов, от которых он зависит. Все пакеты, от которых зависит пакет, должны быть установлены до того, как начнётся его установка. В случае, когда невозможно построить план установки зависимостей, удовлетворяющий данному требованию, установка пакета завершается неудачей и набор зависимостей объявляется некорректным. Никита просит вас реализовать программу, которая будет проверять набор пакетов на наличие некорректных зависимостей.</p> <p><i>Входные данные</i> В первой строке входных данных дано число <math>N</math> — количество пакетов в системе Никиты. В следующих <math>N</math> строках дано описание зависимостей для каждого пакета в виде: <math>D_i \ p_1 \ p_2 \ \dots \ p_{r_i}</math> — первое число — количество пакетов, от которых зависит <math>i</math>-ый пакет, а <math>p_i</math> — номера пакетов.</p> <p><i>Выходные данные</i> В случае если набор зависимостей корректен, выведите единственную строку “Correct” (без кавычек), если же существует какой-либо пакет, попытка установить который завершается неудачей, выведите “Error” (без кавычек).</p> <table border="1"> <thead> <tr> <th>Примеры входных данных</th><th>Примеры выходных данных</th></tr> </thead> <tbody> <tr> <td>2 1 2 1 1</td><td>Error</td></tr> <tr> <td>2 0 1 1</td><td>Correct</td></tr> </tbody> </table> <p>Ожидаемая временная сложность алгоритма: <math>O(N+D_1+D_2+\dots+D_N)</math>.</p> <p>б) Экстремум функции двух переменных. Необходимые и достаточные условия.</p> <p><b>Задача.</b> Найдите наибольшее и наименьшее значение функции</p> $f(x, y) = xy + x + y$ <p>в области <math>D</math>: квадрат <math>1 \leq x \leq 2, \ 2 \leq y \leq 3</math>.</p>	Примеры входных данных	Примеры выходных данных	2 1 2 1 1	Error	2 0 1 1	Correct
Примеры входных данных	Примеры выходных данных						
2 1 2 1 1	Error						
2 0 1 1	Correct						

**Каждое задание в биле представлено в виде двух альтернативных вопросов:**

**а) по информатике**

**или**

**б) по математике.**

**Максимальная оценка 100 баллов ставится за четыре полностью решённых задачи, засчитывается при этом только один из альтернативных вариантов: либо (а), либо (б).**