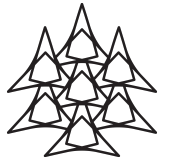
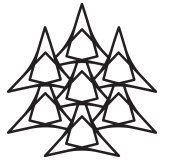
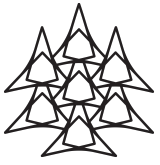


## Серия 1. Системы счисления

1. Переведите в 10-ичную систему числа (Каждый пункт - 0.25 балла)
  - (a)  $10000_2$ ,
  - (b)  $12401_{-7}$ ,
  - (c)  $66666666_7$ ,
  - (d)  $1_2 + 2_4 + 3_8 + \dots + (N^2)_{2N^2}$ ,
  - (e)  $1010101010010_{-2}$
  - (f)  $1221_{2,5}$
2. Представьте число 18437 в системе счисления с основанием  $e$ . Точность - не менее 6 знаков после запятой (1 балл)
3.  $533533_9 = X_{101}$ . Найдите  $X$ . (1 балл)
4. Петя взял число  $N$  и пусть  $S$  - сумма всех таких  $N$ -битных чисел, в двоичной записи которых, единиц не меньше, чем нулей. Докажите, что  $S$  делится на  $2^N - 1$ . (1.5 балла)
5. Докажите, что число  $0.101001000100001000001\dots_2$  не представимо в виде  $\frac{p}{q}$ , где  $p$  - целое, а  $q$  - натуральное. (2 балла)
6. Напишите программу, которая переводит число из системы счисления  $X$  в систему счисления  $Y$ . Формально - даны  $A, X, Y \in \mathbf{N}$ , преобразовать  $A_X$  в  $A_Y$  и вывести. Добавьте в вашу программу проверку корректности записи числа  $A$  в системе счисления  $X$ . Допустимые языки: C, C++, Java, Pascal, JavaScript, Python, Clojure. (2 балла)
7. Дано выражение:  $XYZ_9 + ZXY_{12} + YZX_{15} = 3000_{10}$ . В данном выражении  $X, Y$  и  $Z$  - допустимые различные цифры указанных систем счисления. Найдите какое-нибудь решение выражения или докажите, что решений нет. (2 балла)
8. Найдите все решения уравнения  $(X^2)_Y - N(Y^2)_X = 1$ , где  $X, Y$  - целые числа, а  $N$  - целое число, не являющееся квадратом натурального. (4 балла)



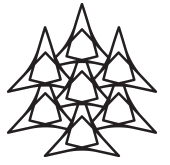
P.S. Нижний индекс обозначает систему счисления. Системы счисления приведены в 10-ичной форме. Если рядом с числом не указан нижний индекс, значит оно находится в 10-ичной системе счисления.



## Серия 2. Логические элементы, булевы функции

1. На вход даются 2 бита  $A_1, B_1$ . Вывести их
  - (a) Сумму отрицаний (0,25 баллов)
  - (b) Штрих Шеффера. (0,25 баллов)
  - (c) Стрелку Пирса. (0,25 баллов)
  - (d) Тожественную единицу (0,25 баллов)
2. На вход даются 3-битные числа  $A_1, A_2, A_3, B_1, B_2, B_3$ . Вывести их сумму (1 балл)
3. На вход даются 3-битные числа  $A_1, A_2, A_3, B_1, B_2, B_3$ . Вывести их произведение (1 балл)
4. На вход даются 3-битные числа  $A_1, A_2, A_3, B_1, B_2, B_3$ . Вывести их сумму по модулю 5. (1,5 балла)
5. Дан логический элемент "Штрих Шеффера". Можно ли выразить через него остальные булевы функции? (2 балла)
6. На вход даются 2 матрицы.  $\begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix} \begin{pmatrix} A_2 & B_2 \\ C_2 & D_2 \end{pmatrix}$ .  
Вывести произведение матриц (2 балла)
7. На вход даются 2 матрицы.  $M_1 = \begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix}, M_2 = \begin{pmatrix} A_2 & B_2 \\ C_2 & D_2 \end{pmatrix}$ .  
Вывести  $\det(M_1)^{\det(M_2)}$ . (2,5 балла)
8. Майнкрафт! Дедлайн - конец лагеря! Соберите 2 битную ячейку памяти, которая была показана на занятии в майнкрафте. (25 баллов)

P.S Матрицы состоят из однобитных чисел. В каждой задаче все вычисления происходят с учётом битов переноса.



### Серия 3. Дробные числа

1. Объясните, почему следующий код выводит false. (1 балл)

```
float f = 1.1f;  
double d = f;  
std::cout « (d == 1.1 ? "true": "false");
```

2. Какое максимальное число unsigned int может храниться во float без потери точности? (1 балл)

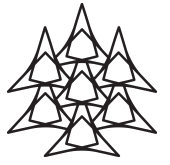
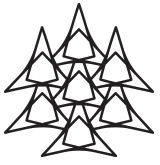
3. Выведите битовое представление числа  $\pi = 3.1415926535$ . (1 балл)

4. Почему 0.1 не может быть точно представлено? Объясните, почему число 0.1 в двоичной системе становится периодической дробью. Приведите примеры других таких чисел. (1 балл)

5. Объясните, почему денормализованные числа нарушают закон тождества ( $x - y == 0 \Rightarrow x == y$ ).

Приведите пример кода, где это проявляется. (1 балл)

6. Возьмите алгоритм численного интегрирования (например, метод трапеций) и исследуйте, как выбор типа (float vs double) влияет на точность результата. (3 балла)



## Серия 4. Затишье перед бурей

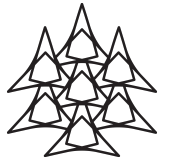
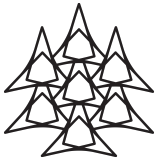
1. Даны арифметические выражения  $V_i$ , записанные в инфиксной записи. Переведите эти выражения в постфиксную запись. Можно заменять выражения на эквивалентные, например  $2 * 2 + 2 * 2 = 2 * (2 + 2)$

- а)  $V_1 : z = 9 * 2 + 13 * 2$ . (0,125 балла)
- б)  $V_2 : z = (187 - 3) * 13$ . (0,125 балла)
- в)  $V_3 : z = 7 * (4 - 1) + 10/2$ . (0,125 балла)
- д)  $V_4 : z = (30/(2 + 3)) * (8 - 4) + 6$ . (0,25 балла)

2. Дано выражение:  $z = 4 * a * b - 13 * a * e + d - c * 13 + 17 * 37$  Архитектура памяти регистрово-регистровая. (0,375 балла)

3.  $z = (a + b) * c + d - e / (f + (g - h * i + j) - k) + (l / m * (n + (o - p)))$ . Дано выражение. Написать команды, которые его вычисляют. Архитектура памяти - аккумуляторная. (1 балл)

4. Напишите программу, которая преобразует выражения, записанные в инфиксной записи в выражения в префиксной записи. Доступные языки: C++, Java. Пример  $(a * b + a / c) \rightarrow (+ * a b / a c)$ . (5 баллов)



## Серия 5. Ассемблер. Синтаксис. Условные операторы

### 1. Учебные задачи

- а) Даны 3 числа, сложите их. (0,5 балла)
- б) Даны 2 числа, проверьте, что первое больше удвоенного второго. (0,5 балла)
- с) Даны 3 числа, сложите их попарные произведения, учитывая переполнение типа данных *uint32\_t*. (0,5 балла)
- д) Даны 2 даты. Сравните их на равенство. (0,5 балла)

2. Даны 3 числа  $a, b, c$ . Положить среднее из  $\frac{a+b}{2}, \frac{b+c}{2}, \frac{a+c}{2}$  в 0-ой регистр. (1 балл)

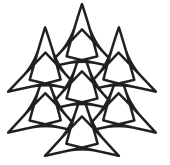
3. Дан год. Выведите в 0-ой регистр, сколько дней в этом году. (Прикол в том, что бывают високосные года) (1 балл)

4. Даны 5 чисел. Отсортируйте их пузырьковой сортировкой. Запишите результат сортировки в первые 5 регистров. (2 балла).

5. Дается шахматная доска. Также даны сначала координаты чёрного слона, потом координаты белого. Через какое минимальное число ходов на доске может остаться ровно 1 слон? Ответ должен быть в первом регистре. (3 балла)

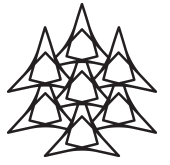
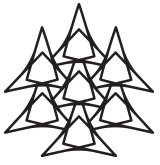
6. Крестики-нолики! Дана доска 3\*3. Клетки нумеруются от 1 до 9. Игроки играют в крестики-нолики. В регистрах лежит информация по каждому ходу в виде 32-битного числа, где первые 8 битов - ASCII-код (0) или (х), ещё по 2 следующих бита на 2 координаты символа на доске. В нулевом регистре должна появиться информация о том, на каком ходу закончилась игра. Также в первом регистре должна появиться информация о результате игры. (7 баллов)

P.S. Вы сами можете определять в каких регистрах будут производиться вычисления и в каких лежат данные.



## Серия 6. Ассемблер. Циклы. Рекурсия

1. Дано число  $N$ . Положите в первый регистр число  $(N!)$  (1 балл)
2. Дано число  $N$ . Найдите  $N$ -ое число Фибоначчи. (1 балл)
3. Даны числа  $a, b$ . Найдите ближайшее к  $b$  число, делящееся на  $a$ . Числа могут быть отрицательными. (1 балл)
4. Дано натуральное число. Определить, сколько в его троичной записи битов, равных 0 или 1. (1 балл)
5. Дано число вида  $2^n 3^k 5^m$ . Вывести список делителей этого числа в разные регистры. (3 балла)
6. Дано число  $N$ . Найдите максимальное такое нечётное  $K$ , что  $K < N^2$  и среди чисел  $N + 1, N + 3, \dots, N + K$  не более 5 простых. (6 баллов)



## Серия 7. Чемпионат по ассемблеру

### Серия 7..1 Правила

В чём состоит игра? В начале у каждой команды есть по 100 баллов. Вам будут даваться некоторые задачи на минимизацию/максимизацию ответа, в которых нужно будет дать только ответ. Раунд происходит следующим образом: сначала все команды ставят по 5 баллов и на доске пишется какой-то неоптимальный ответ. 3 раза по истечении 2 минут можно поднимать ставку. Каждая из команд может поддержать и поставить столько же, поднять, или же может уйти и отдать ставку. В конце оставшиеся в игре команды открывают свои ответы, та команда, у которой ответ оптимальнее, демонстрирует алгоритм решения. Если алгоритм неверный, команда проигрывает и право доказывать решение переходит к следующей команде и так до тех пор, пока одна из команд не докажет ответ. Выигравшая команда забирает на свой счёт очки. Игра заканчивается, когда все команды обанкротятся, или когда задачи закончатся.

### Серия 7..2 Стоимости операций

Рассмотрим стоимости операций:

Операции за 20 тактов: деление (`div r3 r1 r2`), взятие остатка от деления (`rem r3 r1 r2`).

Операции за 10 тактов: добавление переменной (`load r1 x`).

Операции за 4 такта: умножение (`mul r3 r1 r2`).

Операции за 1 такт: загрузка констант (`load r3 const`), сложение (`add/addi r3 r1 r2/const`), вычитание (`sub r3 r1 r2`), битовые операции (`xor/or/and r3 r2 r1/const`).

### Серия 7..3 Задачи

Будут на игре. Эти задачи можно будет сдать только сегодня, за них тоже будут поставлены баллы, возможно частичные, каждому члену команды.