

Задачи с решениями доступны на сайте

1. Написать ассемблерную программу для сравнения двух помеченных чисел (*а* и *б*) в **двойная точность**. Результат работы программы следует найти в реестре *eax* это:

- 0, если это $a = b$
- - 1, если это $a < b$
- 1, если это $a > b$

2. На языке ассемблера реализовать умножение двух чисел с помощью сложения, **использование переменных**

- а. Задача идентична задаче 2 из первого семестра упражнений; единственное отличие состоит в том, что умножаемые два числа, а также результат должны быть помещены в переменные (маркированные ячейки памяти), а не в регистры; числа можно поместить в переменную сразу при инициализации (в части кода `.section .data`), ее не обязательно использовать *двигаться* инструкция
- б. Реализовать оптимизацию как в задаче 2 с первого слагаемого упражнения (минимальное количество итераций цикла)

3. На языке ассемблера реализовать умножение двух чисел с помощью сложения, **использование переменных двойной точности**

- а. Основная идея идентична предыдущей задаче.
- б. Разница: следует использовать 64-битные переменные.
- в. Используйте пример сложения двойной точности, приведенный на слайдах из второго семестра упражнений; проверьте наличие ошибки переполнения

4. Напишите ассемблерную программу, которая находит *n*-й элемент последовательности Фибоначчи, **использование переменных**

- а. Задача идентична задаче 3 из первого семестра упражнений; единственное отличие состоит в том, что число *n*, как и результат, следует помещать в переменные (отмеченные ячейки памяти), а не в регистры; числа можно помещать в переменные прямо при инициализации (в части кода `.section .data`), их не обязательно использовать *двигаться* инструкция
- б. Есть два решения этой задачи; второе решение (оптимизированное) короче и использует меньше регистров

5. Написать ассемблерную программу, которая находит *n*-й элемент последовательности Фибоначчи, **использование переменных двойной точности**

- а. Основная идея идентична предыдущей задаче.
- б. Отличие: учтите, что члены ряда (последовательные сложения Фибоначчи), а также конечный результат 64-битные; число *n* может быть с одинарной точностью

6. Напишите ассемблерную программу, вычисляющую факториал числа *n*, **использование переменных**

- а. Задача идентична задаче 4 из первого семестра занятий; единственное отличие состоит в том, что число *n*, как и результат, следует помещать в переменные (отмеченные ячейки памяти), а не в регистры; числа можно помещать в переменные прямо при инициализации (в части кода `.section .data`), их не обязательно использовать *двигаться* инструкция

7. На языке ассемблера реализовать деление двух чисел с помощью вычитания, сохраняя остаток, **использование переменных двойной точности**

- а. Задача идентична задаче 5 из первого семестра занятий; единственное отличие состоит в том, что делитель, делимое, частное и остаток должны помещаться в переменные (отмеченные ячейки памяти), а не в регистры; числа можно помещать в переменные сразу при инициализации (в части кода `.section .data`), это не обязательно использовать *двигаться* инструкция

Проблемы без решения доступны на сайте

1. Напишите ассемблерную программу, вычисляющую новозеландский доллар из двух чисел: **использование переменных**
 - а. Доработать пример расчета NZD из первого слагаемого упражнений, чтобы для чисел и результата вместо регистров использовались переменные (инициализируются в части кода .section.data, ее не обязательно использовать *двигаться* инструкция начальных значений)
2. В языке ассемблера зарезервируйте одно 32-битное слово и присвойте ему произвольное значение в памяти. После этого его четыре байта помещаются в регистры в порядке **ей, ал, бх, бл**, так что самый младший байт — это **иахи** самый высокий **вбл**
3. Реализовать вычитание в программе на ассемблере **две переменные двойной точности**
 - а. Решите задачу как для отмеченных, так и для немаркированных чисел.
 - б. В качестве подсказки используйте пример сложения в двойной точности со слайдов из второго семестра упражнений.
 - в. Для вычета с использованием кредита **сбб** инструкция
 - д. Обратите внимание на то, вычитание каких частей (выше или ниже) может привести к ошибке перерегулирования.
4. Развернуть задачи 3 и 7 из раздела задач с решениями, чтобы они работали и с отмеченными цифрами.
5. На ассемблере реализуем формулу $a = 2 * b + c$ / $Э$ где они **а, би** числа двойной точности
 - а. Разместить **би** св переменные
 - б. Результат, то есть **а**, тоже должна находиться в памяти, т.е. переменная
6. Расширить задачу 6 из раздела задач с решениями, чтобы она работала с переменными **и двойная точность**
7. Напишите ассемблерную программу, вычисляющую n-ю степень числа b.
 - а. Число **б**, а также **результат**, они должны быть **переменные двойной точности**
 - б. Число является **переменная с одинарной точностью**