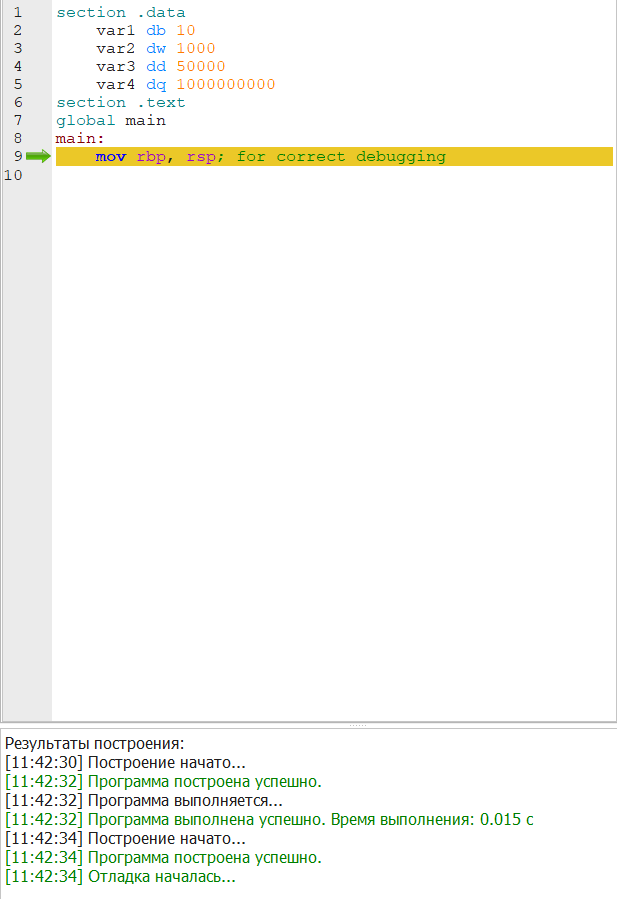
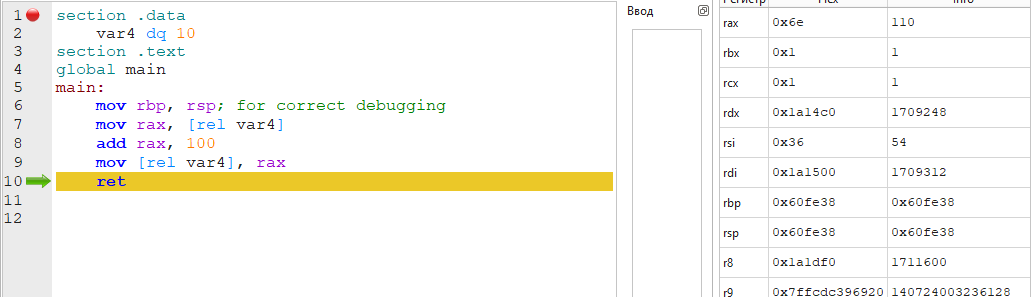
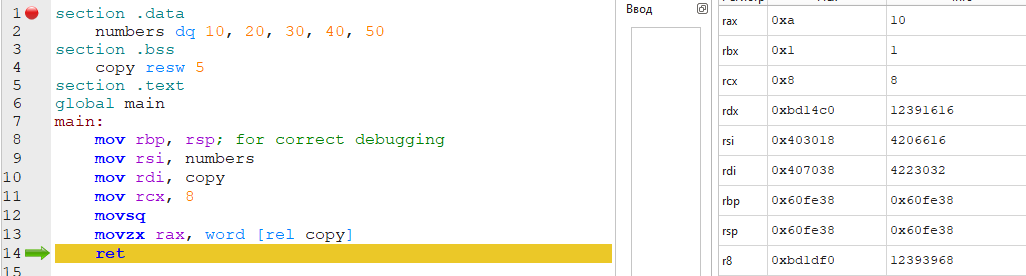
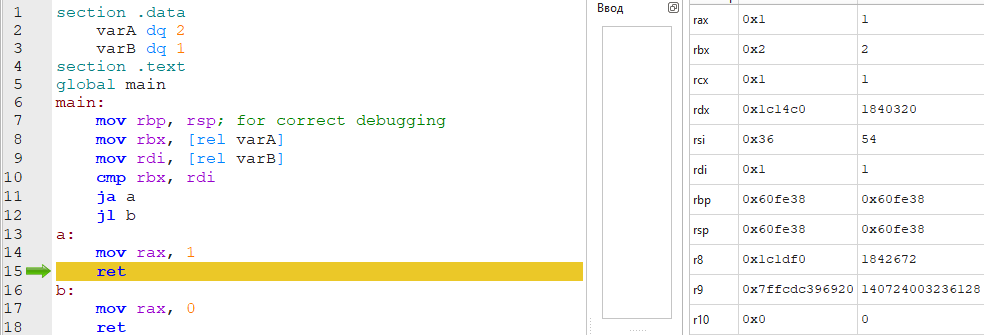
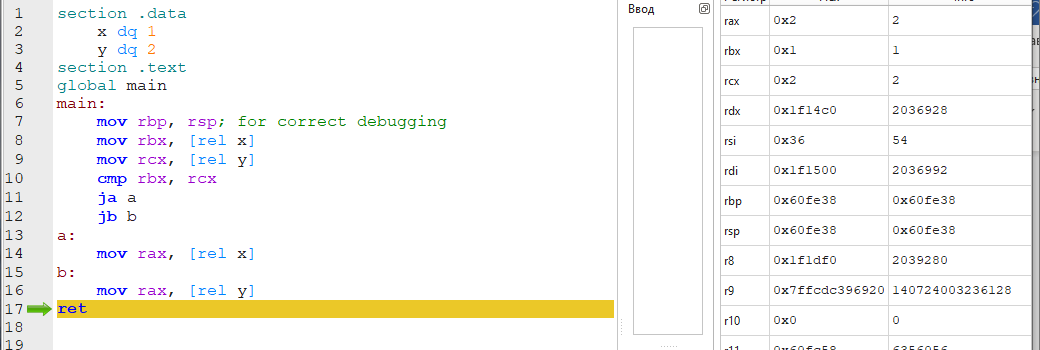
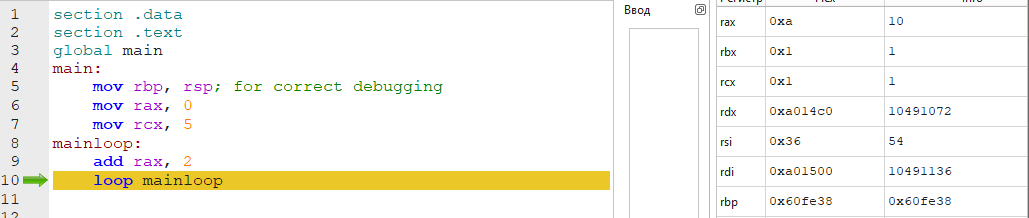
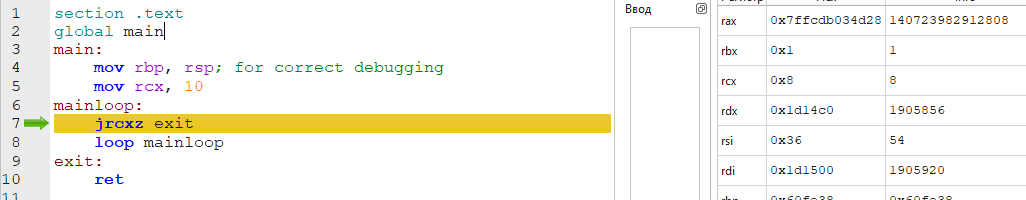
1. 
2. 
3. 
4. 
5. 
6. 
7. 

Вопросы

* 1. *byte: 8-разрядное целое число*
  + *word: 16-разрядное целое число или слово*
  + *dword: 32-разрядное целое число или двойное слово*
  + *qword: 64-разрядное целое число или четверное слово*
* *2. Секция data - инициализированные данные или секция, для чтения и записи, но не для выполнения. Секция bss - секция не инициализированных данных, допустимо чтение данных и запись в секцию.*
* *3*. *Команда CMP процессоров семейства i80x86 используется для сравнения двух операндов. Говоря проще, эта команда сравнивает два числа, то есть проверяет, равны эти два числа или НЕ равны.*
* Эти инструкции влияют только на следующие флаги: CF, ZF, SF, OF, PF, AF
* 4. *jz (или je) – перейти, если нуль или равно. ...*

*jnz (или jne) – перейти, если не нуль или не равно. ...*

* *ja (или jnbe) – перейти, если больше. ...*
* *jae(или jnb) – перейти, если больше или равно. ...*
* *jb (или jnae) – перейти, если меньше. ...*
* *jg (или jnle) – перейти, если больше.\*
* *5. Инструкция****loop****позволяют сократить определение цикла. Она уменьшает на 1 число в регистре RCX и переходит к определенной метке, если RCX не равен нулю. Пример использования на Linux:*
* Выполнение цикла часто зависит от некоторого счетчика - если счетчик равен определенному значению (чаще 0), то выполнение цикла прекращается. Традиционно в качестве подобного регистра-счетчика используется регистр **RCX**. Так, инструкция loop из начала статьи проверяет данный регистр. И для упрощения проверки условия архитектура Intel x86-64 также предоставляет инструкцию **jrcxz** - она проверяет значение RCX, и если оно рано 0, то переходит к определенной метке.