**1. Что такое “ближние” и “дальние” подпрограммы?**

<тип процедуры> - определяет тип перехода: near (близкий), far (дальний). Если тип не задан, по умолчанию принимается near. Тип перехода near показывает, что тело процедуры описано в том же сегменте, что и её вызов. Тип перехода far обеспечивает вызов процедуры из других сегментов, с другим значением регистра CS. Такие процедуры обычно используются как отдельные объектные модули или в составе библиотек.

**2. Как определить “ближний” или “дальний” вариант команды call использован в программе?**call [<тип вызова> ptr] <адрес процедуры>  
< тип вызова > - **near** (**word**) или **far** (**dword**). Если тип не задан, по умолчанию принимается **near**.

**3. Какие способы используются для передачи параметров в подпрограммы?**Для передачи входных параметров в подпрограмму и выходных в программу существует несколько способов. Чаще всего передача параметров осуществляется через регистры или через стек.  
При передаче через регистры перед вызовом подпрограммы параметры заносятся в регистры процессора, а после возврата вызывающая программа забирает из регистров значения результатов.  
При передаче через стек параметры перед вызовом подпрограммы заносятся в стек командой **push src**  
Для обращения к параметрам, хранящимся в стеке, обычно используется регистр bp:

**mov bp, sp**

Необходимо помнить, что поверх параметров, передаваемых в подпрограмму, в стек записываются командой **call** одно или два слова адреса возврата.

Каждая процедура «знает» свой тип вызова (одно или два слова) и отступив от верхушки стека на +2 или на +4 читает параметры.

**mov ax, bp+2** ; первый параметр при ближнем вызове

**mov ax, bp+4** ; первый параметр при дальнем вызове

**4. Может ли массив быть параметром подпрограммы?**

Сам массив – нет, адрес на него – да.

**5. Можно ли использовать для чтения из стека параметров регистр sp вместо bp?**SP, BP - указатель и база стека, соответственно, обеспечивают доступ к данным в стеке, могут использоваться для хранения данных, но делать это не рекомендуется, так как при этом возможно нарушение адресации в стеке, особенно при использовании SP.  
*Проще говоря, можно, но не рекомендуется, т.к. возможны нарушения адресации.*

**6.** **Что означает операнд команды ret?**ret [<выражение>] - выполняет возврат из процедуры в вызывающую программу.  
В зависимости от типа процедуры, эта команда восстанавливает из стека значение IP (ближний вызов) или CS:IP (дальний вызов).

Значение указывает размер стека в байтах, восстанавливаемого при возврате из процедуры. Восстановление стека необходимо производить при передаче параметров процедуры через стек.

Так как работа со стеком выполняется словами, значение всегда должно быть кратным двум.

**7.** **Какой последовательностью команд можно заменить команду “ret 8”?**Удаление 8 байт из стека или четырех слов - (выполнить 8 раз pop), а после вернуться в программу