**ГЛАВА 12**

**1.** Автоматический и регистровый классы и статический без связывания.

**2.** Статический с внешним связыванием, внутренним и без связывания.**7.**

**3.** Статический с внешним связыванием и статический с внутренним связывание.

**4.** Переменные с областью видимости в пределах блока не имеют связывания.

**5.** Слово extern используется для того, чтобы использовать переменную с внешним связыванием или функцию, которые были обьявлены в другом файле программы.

**6.** Функция calloc от malloc отличается тем, что она очищает выделенный блок данных в памяти.

**7.**

int daisy известна main, petal, stem, root.

int lily известна main.

int lily с внутреним связыванием известна функциям в файле 2: stem, root.

int rose с внешним связыванием известна только root, так как, в функции stem есть переменная с таким же названием и она заменяет внешнюю.

**8.**

color в main() равно B

color в first() равно R

color в main() равно B

color в second() равно G

color в main() равно G

**9.**

а) Обьявление статической переменной с областью видимости в пределах файла static int plink говорит о том, что переменную будут использовать разные функции и нужно, чтобы переменная сохраняла свое состояние на протяжении выполнения всей программы.

А прототип функции int value\_ct(const int arr[], int value, int n) говорит о том, что: функция возвращает значание типа данных int;

в качестве формальных параметров принимает адрес первого елемента массива, значения которого не подлежат изминению;

какое-то значение типа int;

и количество элементов в массиве.

б) Защиту значений в вызывающей программе не увеличит, но если добавить квалификатор const к переменной типа int n, то можно избежать ошибки и завершения программы. Так как, если изменить переменную n и сделать ее больше, чем есть элементов в массиве, то можно залезть в блок памяти, который используется другой программой.