1. В основном операционная система использует адресное пространство для того, чтобы обрабатывать несколько процессов одновременно. Адресное пространство делится на следующим образом: код – область памяти, где находится код скомпилированной программы, стек – область, где программа хранит информацию о функциях, аргументах и локальных переменных этой фикции, куча – область, находящая под контролем программиста (может делать все, что угодно), свободное место – свободная область, которая может быть занята стеком и/или кучей
2. В плюсах массивы и указатели тесно связаны. С помощью указателей мы также легко можем манипулировать элементами массива, как и с помощью индексов. Имя массива без индексов в Си ++ является адресом его первого элемента. Соответственно через операцию разыменования мы можем получить значение по этому адресу. Мы можем пробежаться по всем элементом массива, прибавляя к адресу определенное число. Например, если к адресу прибавим 2, то получим 1+2 = 3 элемент и так далее
3. Одна из главных опасностей низкоуровневой работой с памятью является ее захламление. При низкоуровневой работе все больше обязанностей возлагаются на программиста, а не на язык программирования, в данном случае если программисту потребуется выделить память низкоуровневыми командами, тогда он также должен освободиться эту память, иначе она останется занятой и не освободится при окончании работы программы.
4. -
5. Данные в функцию можно передать 3-мя способами: по значению (фактический параметр вычисляется в вызванной функции и это значение передается на место этого параметра), по адресу (Способ передачи параметров по адресу используется для того, чтобы через указатели обеспечить непосредственный доступ к значениям аргументов), и по ссылке (при передаче параметров по ссылке передается ссылка на объект, через которую мы можем манипулировать самим объектов, а не просто его значением)