Арифметика указателей приводит к следующему: a[i]= \*(a+i) = \*(i+a) = i[a]

Т.е. в записи кода можно менять местами имя массива и значения индекса элемента в квадратных скобках. Т.е. записи a[3] и 3[a] равноценны. поясню на примере, например есть массив int a[] = { 1,3,5,7,9,11 }; тогда a[3]=7 .

Если записать в коде 3[a] то тоже получим 7 Это приводит к парадоксальным записям кода, которые программисты не используют, но на собеседованиях их могут попросить пояснить. Например, задачка: что выведется на экран из такого : int a[] = { 4,3,5,7,9,11 }; int a2 = 3[a+1]-a[5]+(a-1)[4]; cout << a2 << endl;

Ответ: 5 логика будет такой: 1) 3[a+1] - имя массива 'а' это указатель на начальный элемент, то есть на четверку, а+1 сдвинет указатель начала на следующий элемент массива, т.е на тройку, получится {3,5,7,9,11}, тогда 3[a+1] указывает на девятку 2) a[5] = 11, здесь все привычно и ничего не меняется, это просто . 3) (a-1)[4] - здесь разность в круглых скобках сдвинет указатель на 1 элемент влево, то есть получится новый массив с каким то мусором в начале {-858993460,1,3,5,7,9,11 }, а элемент с индексом 4 такого массива это 7 . Поэтому на экран и выведется 5, так как а2= 9 - 11 + 7 = 5

