

Mar 26, 23 20:14	indxoffrm.s	Page 1/2
<pre> /**  * indxoffrm - Returns the index of the first occurrence of character ch <b>in</b> the  * given string  *             after the specified index "fromIndex".  *  * @param x0: Address of the target string to search.  * @param x1: Character to find <b>in</b> the target string.  * @param x2: Index to start the search <b>in</b> the target string.  *  * @return x0: Integer index of the first occurrence of character ch <b>in</b> the targ  * et string  *             after the specified index "fromIndex". If the character is not fou  * nd,  *             returns -1.  *  * This <b>function</b> searches <b>for</b> the first occurrence of the specified character ch  * <b>in</b> the  * target string starting from the specified index "fromIndex" onwards. The <b>funct  * ion</b> returns  * the index of the first occurrence of the character ch <b>in</b> the target string af  * ter the  * specified index "fromIndex". The search starts from the index x2+1 <b>in</b> the targ  * et string,  * where x2 is the specified index "fromIndex". The <b>function</b> returns -1 <b>if</b> the ch  * aracter ch  * is not found <b>in</b> the target string.  *  * Registers used: x0, x1, x2, w2  * Registers saved: none  */  .text .global indxoffrm indxoffrm:     // Save the target string pointer on the stack     str x0, [sp, #-16]!      // Calculate the starting index <b>for</b> the search     add x0, x0, x2     add x0, x0, #1  loop:     // Load a byte from the target string     ldrb w2, [x0], #1      // Check <b>if</b> the byte is the terminating null byte     cmp w2, #0     b.eq error      // Check <b>if</b> the byte is the character to find     cmp w2, w1     b.eq found      // Jump back to the beginning of the loop     b loop  found:     // Restore the target string pointer from the stack     ldr x1, [sp], #16      // Calculate and <b>return</b> the index of the found character     sub x0, x0, x1 </pre>		

Mar 26, 23 20:14	indxoffrm.s	Page 2/2
<pre>     sub x0, x0, #1     ret  error:     // Remove the saved target string pointer from the stack     add sp, sp, #16      // Return -1 to indicate that the character was not found     mov x0, #-1     ret </pre>		