

Programming Language Subset Summary of changes Summer 2023

Pearson Edexcel Level 1/Level 2 GCSE (9–1) in
Computer Science Paper 2 – Application of
Computational Thinking (1CP2/02)

Introduction

In this document, teachers and centres are informed about the changes made to the programming language subset (PLS) document from summer 2022 to summer 2023.

The programming language subset (PLS) document version 4 for Summer 2023 can be found [here](#).

Please see list of changes below.

Version 4 (Summer 2023)	Version 2 (Summer 2022)								
Iteration – Page 7 Arguments to range() added for completeness	Iteration – Page 7								
<table><tr><td>for <id> in range (<start>, <stop>): <command></td><td>Count-controlled loop. Executes <command> a fixed number of times, based on the numbers generated by the range function. <stop> is required. <start> is optional.</td></tr><tr><td>for <id> in range (<start>, <stop>, <step>): <command></td><td>Same as above, except that <step> influences the numbers generated by the range function. <stop> is required. <start> and <step> are optional.</td></tr></table>	for <id> in range (<start>, <stop>): <command>	Count-controlled loop. Executes <command> a fixed number of times, based on the numbers generated by the range function. <stop> is required. <start> is optional.	for <id> in range (<start>, <stop>, <step>): <command>	Same as above, except that <step> influences the numbers generated by the range function. <stop> is required. <start> and <step> are optional.	<table><tr><td>for <id> in range (<start>, <stop>): <command></td><td>Count-controlled loop. Executes <command> a fixed number of times, based on the numbers generated by the range function</td></tr><tr><td>for <id> in range (<start>, <stop>, <step>): <command></td><td>Same as above, except that <step> influences the numbers generated by the range function</td></tr></table>	for <id> in range (<start>, <stop>): <command>	Count-controlled loop. Executes <command> a fixed number of times, based on the numbers generated by the range function	for <id> in range (<start>, <stop>, <step>): <command>	Same as above, except that <step> influences the numbers generated by the range function
for <id> in range (<start>, <stop>): <command>	Count-controlled loop. Executes <command> a fixed number of times, based on the numbers generated by the range function. <stop> is required. <start> is optional.								
for <id> in range (<start>, <stop>, <step>): <command>	Same as above, except that <step> influences the numbers generated by the range function. <stop> is required. <start> and <step> are optional.								
for <id> in range (<start>, <stop>): <command>	Count-controlled loop. Executes <command> a fixed number of times, based on the numbers generated by the range function								
for <id> in range (<start>, <stop>, <step>): <command>	Same as above, except that <step> influences the numbers generated by the range function								

Version 4 (Summer 2023)	Version 2 (Summer 2022)																												
<p>Built-in Subprograms Page 9</p> <p>Bool, float, int and str added for completeness</p> <table> <tr> <th>Subprogram</th><th>Description</th></tr> <tr> <td>bool (<item>)</td><td>Returns <item> converted to the equivalent Boolean value</td></tr> <tr> <td>chr (<integer>)</td><td>Returns the string which matches the Unicode value of <integer>. The first 128 characters of Unicode are equivalent to ASCII.</td></tr> <tr> <td>float (<item>)</td><td>Returns <item> converted to the equivalent real value</td></tr> <tr> <td>int (<item>)</td><td>Returns <item> converted to the equivalent integer value</td></tr> <tr> <td>ord (<char>)</td><td>Returns the integer equivalent to the Unicode string of the single character <char>. The first 128 characters of Unicode are equivalent to ASCII.</td></tr> <tr> <td>range (<start>, <stop>, <step>)</td><td>Generates a list of numbers using <step>, beginning with <start> and up to, but not including, <stop>. A negative value for <step> goes backwards. <stop> is required. <start> and <step> are optional. The default value for <start> is zero. The default value for <step> is positive one.</td></tr> <tr> <td>round (<x>, <n>)</td><td>Rounds <x> to the number of <n> digits after the decimal (uses the 0.5 rule). The <n> is optional. If omitted, the function returns the nearest integer to <x>.</td></tr> <tr> <td>str (<item>)</td><td>Returns <item> converted to the equivalent string value</td></tr> </table>	Subprogram	Description	bool (<item>)	Returns <item> converted to the equivalent Boolean value	chr (<integer>)	Returns the string which matches the Unicode value of <integer>. The first 128 characters of Unicode are equivalent to ASCII.	float (<item>)	Returns <item> converted to the equivalent real value	int (<item>)	Returns <item> converted to the equivalent integer value	ord (<char>)	Returns the integer equivalent to the Unicode string of the single character <char>. The first 128 characters of Unicode are equivalent to ASCII.	range (<start>, <stop>, <step>)	Generates a list of numbers using <step>, beginning with <start> and up to, but not including, <stop>. A negative value for <step> goes backwards. <stop> is required. <start> and <step> are optional. The default value for <start> is zero. The default value for <step> is positive one.	round (<x>, <n>)	Rounds <x> to the number of <n> digits after the decimal (uses the 0.5 rule). The <n> is optional. If omitted, the function returns the nearest integer to <x>.	str (<item>)	Returns <item> converted to the equivalent string value	<p>Built-in Subprograms Page 9</p> <table> <tr> <th>Subprogram</th><th>Description</th></tr> <tr> <td>chr (<integer>)</td><td>Returns the string which matches the Unicode value of <integer></td></tr> <tr> <td>ord (<char>)</td><td>Returns the integer equivalent to the Unicode string of the single character <char></td></tr> <tr> <td>range (<start>, <stop>, <step>)</td><td>Generates a list of numbers using <step>, beginning with <start> and up to, but not including, <stop>. A negative value for <step> goes backwards</td></tr> <tr> <td>round (<x>, <n>)</td><td>Rounds <x> to the number of <n> digits after the decimal (uses the 0.5 rule)</td></tr> </table>	Subprogram	Description	chr (<integer>)	Returns the string which matches the Unicode value of <integer>	ord (<char>)	Returns the integer equivalent to the Unicode string of the single character <char>	range (<start>, <stop>, <step>)	Generates a list of numbers using <step>, beginning with <start> and up to, but not including, <stop>. A negative value for <step> goes backwards	round (<x>, <n>)	Rounds <x> to the number of <n> digits after the decimal (uses the 0.5 rule)
Subprogram	Description																												
bool (<item>)	Returns <item> converted to the equivalent Boolean value																												
chr (<integer>)	Returns the string which matches the Unicode value of <integer>. The first 128 characters of Unicode are equivalent to ASCII.																												
float (<item>)	Returns <item> converted to the equivalent real value																												
int (<item>)	Returns <item> converted to the equivalent integer value																												
ord (<char>)	Returns the integer equivalent to the Unicode string of the single character <char>. The first 128 characters of Unicode are equivalent to ASCII.																												
range (<start>, <stop>, <step>)	Generates a list of numbers using <step>, beginning with <start> and up to, but not including, <stop>. A negative value for <step> goes backwards. <stop> is required. <start> and <step> are optional. The default value for <start> is zero. The default value for <step> is positive one.																												
round (<x>, <n>)	Rounds <x> to the number of <n> digits after the decimal (uses the 0.5 rule). The <n> is optional. If omitted, the function returns the nearest integer to <x>.																												
str (<item>)	Returns <item> converted to the equivalent string value																												
Subprogram	Description																												
chr (<integer>)	Returns the string which matches the Unicode value of <integer>																												
ord (<char>)	Returns the integer equivalent to the Unicode string of the single character <char>																												
range (<start>, <stop>, <step>)	Generates a list of numbers using <step>, beginning with <start> and up to, but not including, <stop>. A negative value for <step> goes backwards																												
round (<x>, <n>)	Rounds <x> to the number of <n> digits after the decimal (uses the 0.5 rule)																												

Version 4 (Summer 2023)	Version 2 (Summer 2022)												
String Subprograms Page 11 Top two only, detail added	String Subprograms Page 10												
<table> <tr> <th>Subprogram</th><th>Description</th></tr> <tr> <td><code><string>.find (<substring>, <start>, <end>)</code></td><td>Returns the location of the first instance of <code><substring></code> in the original <code><string></code>, reading from left to right. <code><start></code> is the index to begin the find. The default is zero. <code><end></code> is the index to stop the find. The default is the end of the string. Returns -1, if not found.</td></tr> <tr> <td><code><string>.index (<substring>, <start>, <end>)</code></td><td>Returns the location of the first instance of <code><substring></code> found in the original <code><string></code> as read from left to right. Raises an exception if not found. <code><substring></code> is required. <code><start></code> and <code><end></code> are optional. The default value for <code><start></code> is zero. The default value for <code><end></code> is the end of the string.</td></tr> </table>	Subprogram	Description	<code><string>.find (<substring>, <start>, <end>)</code>	Returns the location of the first instance of <code><substring></code> in the original <code><string></code> , reading from left to right. <code><start></code> is the index to begin the find. The default is zero. <code><end></code> is the index to stop the find. The default is the end of the string. Returns -1, if not found.	<code><string>.index (<substring>, <start>, <end>)</code>	Returns the location of the first instance of <code><substring></code> found in the original <code><string></code> as read from left to right. Raises an exception if not found. <code><substring></code> is required. <code><start></code> and <code><end></code> are optional. The default value for <code><start></code> is zero. The default value for <code><end></code> is the end of the string.	<table> <tr> <th>Subprogram</th><th>Description</th></tr> <tr> <td><code><string>.find (<substring>)</code></td><td>Returns the location of <code><substring></code> in the original <code><string></code>. Returns -1, if not found</td></tr> <tr> <td><code><string>.index (<substring>)</code></td><td>Returns the location of <code><substring></code> in the original <code><string></code>. Raises an exception if not found</td></tr> </table>	Subprogram	Description	<code><string>.find (<substring>)</code>	Returns the location of <code><substring></code> in the original <code><string></code> . Returns -1, if not found	<code><string>.index (<substring>)</code>	Returns the location of <code><substring></code> in the original <code><string></code> . Raises an exception if not found
Subprogram	Description												
<code><string>.find (<substring>, <start>, <end>)</code>	Returns the location of the first instance of <code><substring></code> in the original <code><string></code> , reading from left to right. <code><start></code> is the index to begin the find. The default is zero. <code><end></code> is the index to stop the find. The default is the end of the string. Returns -1, if not found.												
<code><string>.index (<substring>, <start>, <end>)</code>	Returns the location of the first instance of <code><substring></code> found in the original <code><string></code> as read from left to right. Raises an exception if not found. <code><substring></code> is required. <code><start></code> and <code><end></code> are optional. The default value for <code><start></code> is zero. The default value for <code><end></code> is the end of the string.												
Subprogram	Description												
<code><string>.find (<substring>)</code>	Returns the location of <code><substring></code> in the original <code><string></code> . Returns -1, if not found												
<code><string>.index (<substring>)</code>	Returns the location of <code><substring></code> in the original <code><string></code> . Raises an exception if not found												