



Cambridge International AS & A Level

COMPUTER SCIENCE

9618/21

Paper 2 Fundamental Problem-solving and Programming Skills

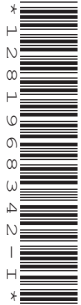
May/June 2023

INSERT

2 hours

INFORMATION

- This insert contains all the resources referred to in the questions.
- You may annotate this insert and use the blank spaces for planning. **Do not write your answers** on the insert.



This document has **4** pages.

An error will be generated if a function call is not properly formed or if the parameters are of an incorrect type or an incorrect value.

String and Character Functions

- A string of length 1 may be either of type CHAR or STRING
- A CHAR may be assigned to, or concatenated with, a STRING
- A STRING of length greater than 1 cannot be assigned to a CHAR

<code>LEFT(ThisString : STRING, x : INTEGER) RETURNS STRING</code> returns leftmost <code>x</code> characters from <code>ThisString</code> Example: <code>LEFT("ABCDEFGH", 3)</code> returns "ABC"
<code>RIGHT(ThisString : STRING, x : INTEGER) RETURNS STRING</code> returns rightmost <code>x</code> characters from <code>ThisString</code> Example: <code>RIGHT("ABCDEFGH", 3)</code> returns "FGH"
<code>MID(ThisString : STRING, x : INTEGER, y : INTEGER) RETURNS STRING</code> returns a string of length <code>y</code> starting at position <code>x</code> from <code>ThisString</code> Example: <code>MID("ABCDEFGH", 2, 3)</code> returns string "BCD"
<code>LENGTH(ThisString : STRING) RETURNS INTEGER</code> returns the integer value representing the length of <code>ThisString</code> Example: <code>LENGTH("Happy Days")</code> returns 10
<code>TO_UPPER(x : <datatype>) RETURNS <datatype></code> <datatype> may be CHAR or STRING returns an object of type <datatype> formed by converting all characters of <code>x</code> to upper case. Examples: • <code>TO_UPPER("Error 803")</code> returns "ERROR 803" • <code>TO_UPPER('a')</code> returns 'A'
<code>TO_LOWER(x : <datatype>) RETURNS <datatype></code> <datatype> may be CHAR or STRING returns an object of type <datatype> formed by converting all characters of <code>x</code> to lower case. Examples: • <code>TO_LOWER("JIM 803")</code> returns "jim 803" • <code>TO_LOWER('W')</code> returns 'w'
<code>NUM_TO_STR(x : <datatype1>) RETURNS <datatype2></code> returns a string representation of a numeric value. <datatype1> may be REAL or INTEGER, <datatype2> may be CHAR or STRING Example: <code>NUM_TO_STR(87.5)</code> returns "87.5"
<code>STR_TO_NUM(x : <datatype1>) RETURNS <datatype2></code> returns a numeric representation of a string. <datatype1> may be CHAR or STRING, <datatype2> may be REAL or INTEGER Example: <code>STR_TO_NUM("23.45")</code> returns 23.45
<code>IS_NUM(ThisString : <datatype>) RETURNS BOOLEAN</code> returns TRUE if <code>ThisString</code> represents a valid numeric value. <datatype> may be CHAR or STRING Example: <code>IS_NUM("-12.36")</code> returns TRUE

ASC(ThisChar : CHAR) RETURNS INTEGER

returns an integer value (the ASCII value) of character ThisChar

Example: ASC('A') returns 65, ASC('B') returns 66, etc.

CHR(x : INTEGER) RETURNS CHAR

returns the character whose integer value (the ASCII value) is x

Example: CHR(65) returns 'A', CHR(66) returns 'B', etc.

Numeric Functions

INT(x : REAL) RETURNS INTEGER

returns the integer part of x

Example: INT(27.5415) returns 27

RAND(x : INTEGER) RETURNS REAL

returns a real number in the range 0 to x (**not** inclusive of x).

Example: RAND(87) may return 35.43

Date Functions

Date format is assumed to be DD/MM/YYYY unless otherwise stated.

DAY(ThisDate : DATE) RETURNS INTEGER

returns the current day number from ThisDate

Example: DAY(04/10/2003) returns 4

MONTH(ThisDate : DATE) RETURNS INTEGER

returns the current month number from ThisDate

Example: MONTH(04/10/2003) returns 10

YEAR(ThisDate : DATE) RETURNS INTEGER

returns the current year number from ThisDate

Example: YEAR(04/10/2003) returns 2003

DAYINDEX(ThisDate : DATE) RETURNS INTEGER

returns the day index number from ThisDate where Sunday = 1, Monday = 2 etc.

Example: DAYINDEX(09/05/2023) returns 3

SETDATE(Day, Month, Year : INTEGER) RETURNS DATE

returns a value of type DATE with the value of <Day>/<Month>/<Year>

Example: SETDATE(26, 10, 2003) returns a date corresponding to 26/10/2003

TODAY() RETURNS DATE

returns a value of type DATE corresponding to the current date.

Text File Functions

EOF(FileName : STRING) RETURNS BOOLEAN

returns TRUE if there are no more lines to be read from file FileName

The function will generate an error if the file is not already open in READ mode.

Operators

An error will be generated if an operator is used with a value or values of an incorrect type.

&	concatenates (joins) two strings. Example: "Summer" & " " & "Pudding" evaluates to "Summer Pudding" The operator may also be used to concatenate a CHAR with a STRING
AND	performs a logical AND on two Boolean values. Example: TRUE AND FALSE evaluates to FALSE
OR	performs a logical OR on two Boolean values. Example: TRUE OR FALSE evaluates to TRUE
NOT	performs a logical NOT on a Boolean value. Example: NOT TRUE evaluates to FALSE
MOD	finds the remainder when one number is divided by another. Example: 10 MOD 3 evaluates to 1
DIV	finds the quotient when one number is divided by another. Example 10 DIV 3 evaluates to 3

Comparison Operators

=	used to compare two items of the same type. evaluates to TRUE if the condition is true, otherwise evaluates to FALSE
>	Notes:
<	- may be used to compare types REAL and INTEGER - may be used to compare types CHAR and STRING - case sensitive when used to compare types CHAR or STRING - cannot be used to compare two records
>=	
<=	Examples:
<>	"Program" = "program" evaluates to FALSE - Count = 4 evaluates to TRUE when variable Count contains the value 4

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and cleared where possible. Every reasonable effort has been made by the publisher (UCLES) to trace copyright holders, but if any items requiring clearance have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced online in the Cambridge Assessment International Education Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cambridgeinternational.org after the live examination series.

Cambridge Assessment International Education is part of Cambridge Assessment. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which is a department of the University of Cambridge.