

```

# MacVersion = 1.0
# MacDescription = Scan through cache list and identify trails/series (Using code adapted from CacheSeries.gsk) then
# add the matched part of the trail name to UserData2, and the number of caches in the trail to UserData3.
# You can then easily sort and filter on trails and number of caches in trails.
# MacAuthor = GeoCass
# MacFileName = CacheTrails.gsk
# MacURL =

#####
#          THIS SECTION TAKEN FROM CacheSeries.gsk (MacURL = http://gsak.net/board/index.php?showtopic=958)          #
#####
$Seriesfile = ":memory:"
$status = DbToSqlite("caches", "Name, Code", $Seriesfile)

$match = "14" # Number of characters to match in a name
$members = "4" # Minimum number of caches in a trail to count it as a trail

GOSUB name=SearchCaches
BEGINSUB name=SearchCaches

    # Amended to exclude anything starting 'Church Micro' as we have a lot of these in the UK that aren't trails
    $_sql = "CREATE TABLE series AS SELECT count(name) as c, substr(name, 1, $match) as name2 FROM caches WHERE
(name2 NOT LIKE 'Church Micro%') and (name2 NOT LIKE 'SideTracked%') GROUP BY substr(name, 1, $match) COLLATE NOCASE
HAVING c>=$members ORDER BY c DESC"
    $status = Sqlite("sql", $_sql)

    $_sql = "SELECT * FROM series"
    $status = Sqlite("sql", $_sql)

    $count = $_SqlRows

    $_sql = "INSERT INTO series SELECT count(name) as c, substr(name, -$match, $match) as name2 FROM caches GROUP BY
substr(name, -$match, $match) COLLATE NOCASE HAVING c>=$members ORDER BY c DESC"
    $status = Sqlite("sql", $_sql)

    $_sql = "SELECT * FROM series"
    $status = Sqlite("sql", $_sql)
    $count = $_SqlRows
                                #Revise the value in $count for use in the array
#####
# Scan the values returned from the series table and replace any end of lines (\r\n) with semicolons.
# Now all numbers and names are separated with semicolons and we can easily strip each bit out of the string

```

```

$status=RegexReplace("\r\n",$status,";")
$status=RegexReplace("\(", $status, "")
$status=RegexReplace("\)", $status, "")

# Initialize the arrays to store the returned values. Array 1 stores the count, Array 2 stores the names
# The length of the arrays are taken from the variable storing the number of rows in the table
# The elements in the arrays match each other, so the number in Array(1,9) matches the name in Array(2,9)
Array(1,0) = "$count" # Trail count
Array(2,0) = "$count" # Name of trail
$status=$status+";" #End the final name with a semicolon, so it can be easily split

# We're going to use a loop to go through each one of the trails and store the numbers in array 1 and the
# names in array2. We do this by extracting everything up to a semicolon from the first position, and then
# everything up to a semicolon from the position after it. We then move on to the next trail.
$loopcount = 1
$poscount = 1
WHILE ($loopcount <= $count)
    Array(1,$loopCount)=Extract($status, ";", $poscount)
    Array(2,$loopCount)=Extract($status, ";", $poscount+1)
    $loopcount=$loopcount+1
    $poscount=$poscount+2
ENDWHILE

# Remove the temporary series table
$_sql = "DROP TABLE IF EXISTS series"
$status = Sqlite("sql",$_sql)

# Set loopcount back to 1 and use it again for this loop
$loopcount = 1
# Loop through the number of rows (i.e. the number of elements in each array)
WHILE ($loopcount <= $count)
    GOTO Position=Top

    WHILE not($_EOL) # Loop through the entire file
        # Match the cache name with the matched name in the array
        IF Regex(Array(2,$loopCount), $d_Name )
            # Update UserData to Trail, UserData2 to the matched trail name
            $d_User2= Array(2,$loopCount)

            # The user data fields can only contain string values. To enable these to be sorted numerically this IF
statement

```

```
# adds a '0' in front of any of the numbers that are less than 10. This way numbers under 10 display as
# "01", "02", "03", etc.
$UnderTen = addstr(1, "clear")
IF (Len(Array(1, $loopCount)) < 2)
    $UnderTen = addstr(1, "add", "0" + Array(1, $loopCount))
    $UnderTen = addstr(1, "get")
    $d_User3=$UnderTen
ELSE
    $d_User3=Array(1, $loopCount) # If number bigger than 10, don't change it
ENDIF
ENDIF
GOTO Position=Next
ENDWHILE
$loopCount = $loopcount + 1
ENDWHILE
ENDSUB
```