
Article

[Danny Wijnschenk](#) · Nov 5, 2017 4m read

Advent of Code 2016 Day5: How about a Nice Game of Chess?

This is a series of programming challenges for beginners and experienced Caché programmers.

For an introduction : goto to article <https://community.intersystems.com/post/advent-code-2016-day1-no-time-ta...>

The challenge of day 5 is to calculate a password of 8 characters by finding the MD5 hash of the input and an increasing integer index.

The password is constructed by taking the 6th character of the first 8 hashes that start with 5 zeroes (in hex representation).

For example : if the input would be 'abc', the first hash that has 5 zeroes would be by MD5 hashing abc3231929 : the hex representation is 00 00 01 55 F8 ...

The 6th character is 1 and thus the first character of the password.

The full explanation and instructions of this challenge can be found on <http://adventofcode.com/2016/day/5>, you will also find your input hash on that page. (the input is different for each logged in user at the adventofcode site).

Luckily, Caché has a class with the most common encryption methods : \$System.Encryption (More documentation can be found on <http://docs.intersystems.com/latest/csp/documatic/%25CSP.Documatic.cls?C...>) which includes MD5Hash.

The output of this method is a 16-byte MD5 hash, which we need to convert to hexadecimal representation using \$ZHex() of each decimal character.

Here is the code Bert wrote to solve this challenge :

```
Start() PUBLIC {
    set doorID="reyedfim"
; "uqwqemis" //each user of the AOC website will get other input
    set id=0
    set isDone=0
    set result=""
    while ('isDone) {
        set id=id+1
        set hash=##class(%SYSTEM.Encryption).MD5Hash(doorID_id)
        set a=$ascii($extract(hash,1))
        set b=$ascii($extract(hash,2))
        set c=$ascii($extract(hash,3))
        if ((a=0) && (b=0) && (c<=15)) {
            set result=result_$ZHEX(c)
            if $length(result)=8 {
                set isDone=1
            }
        }
    }
    write !,result
}
```

After submitting the right answer on the adventofcode website, it unlocks the second part of the challenge : instead of taking the 6th character for the password, we have to take the 7th character for that, and use the 6th character for the position in the password. Only if the position is between 0 and 7 (and not already found) the character will be used in the password.

To be proud of your solution, you have to give an output which looks like cinematic "decrypting" animation, often used in movies when a cypher is about to be un-cyphered like

<https://www.youtube.com/watch?v=NHWjICaIrQo&t=25> (This movie came out when i started to program!)

I simply output the unfinished password with a carriage return instead of CR/LF (Write \$Char(13) instead of Write !) , and turned the text cursor off. I had to limit the output to every 1000 hashes, because the full output was slowing down the code too much.

Here is my solution :

```
ClassMethod Part2()
{
    #Dim input as %String = "reyedfim"
    #Dim password,display as %String = " "
    #Dim hash, md5Hash, hex, hex1 as %String
    #Dim index, iHash as %Integer

    Write $Char(27),"[?25l" //Hide the cursor for nicer output effect
    Try {
        For index = 1:1 {
            Set hash = input_index
            Set md5Hash = $System.Encryption.MD5Hash(hash)
            Set hex=""
            For iHash=1:1:$l(md5Hash) {
                Set hex1 = $ZHex($A(md5Hash,iHash))
                If $Length(hex1)=1 set hex1="0"_hex1
                Set hex = hex _ hex1
            }
            If $E(hex,1,5)="00000" {
                If ($A(hex,6)<$A("0"))!($A(hex,6)>$A("7")) Continue
                If $E(password,$E(hex,6)+1)=" " {
                    Set $E(password,$E(hex,6)+1) = $E(hex,7)
                    Set $E(display,$E(hex,6)+1) = $E(hex,7)
                }
                If password'[" " Quit //found complete password
            } else {
                //wrong hash, but we want to use it in our display to rotate digits that are not yet found
                If ($A(hex,6)'<$A("0"))&($A(hex,6)'>$A("7")) {
                    If $E(password,$E(hex,6)+1)=" " Set $E(display,$E(hex,6)+1) = $E(hex,7)
                }
            }
            If index#1000=0 Write $Char(13),display," "
            //to display a nice cinematic decrypting animation
        }
    } Catch {
        Use 0 Write "Error : ", $ZError,!
    }
    Write $Char(27),"[?25h" //turn cursor on again
    Write !,"password = ",password,!
    Quit
}
```

Look here for the solutions on all the 2016 challenges as we move along on the advent :

<https://github.com/DannyWijnschenk/AdventOfCode2016> and <https://bitbucket.org/bertsarens/advent2016>.

Here is the list of all Advent of Code 2016 articles :

[1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) [12](#) [13](#) [14](#) [15](#) [16](#) [17](#) [18](#) [19](#) [20](#) [21](#) [22](#) [23](#) [24](#) [25](#)

[#Caché](#) [#Code Snippet](#) [#Contest](#) [#ObjectScript](#)

Source URL: <https://community.intersystems.com/post/advent-code-2016-day5-how-about-nice-game-chess>