

---

Toontrack.EZkeys.FuLL.VSTi.AAX.RTAS.v1.2.4.WIN Free Download NEW!



---

Category:Toontrack Software Category:Sample-based audio softwareQ: How do I compare hash size with a fixed amount? I have an array of hashes where each hash contains a key and a value. There is this function to find a hash with a given value sub findhash( my(%hash, \$value) = @\_; return undef unless @\_ > 1; my \$one = pop; my \$two = \$one; for ( \$two->{ \$value } ) { print "Sone!\t\$two!\t\$hash{\$one}\t\$value "; \$one = \$two; \$two = \$one; } return \$two if exists \$hash{\$one}; return undef; } Now I want to find the hash that has a size of 5, the correct one is \$hash{\$one} = \$two; The size should be one, because \$two contains the correct value. How to do this? It does not have to be a smart way, since I am just comparing if a hash has a size of 5, all the other parts of the program is irrelevant. (I am using perl 5.10) A: For example: sub find\_hash\_with\_size { my ( \$size, %h ) = @\_; return undef unless @\_ > 1; my %h\_copy = %h; for ( 1..\$size ) { \$h\_copy{ \$\_ } = () for keys %h; } return \$h\_copy; } Then use it like this: find\_hash\_with\_size( 5, \%h ); EDIT: The more efficient way would be to avoid using an array reference and to use the pop with the hash key instead, like this: sub find\_hash\_with\_size { my ( \$size, %h ) = @

