##C++CL.hs.CLHsNT.NSCODES AGAINST AN ANTI-354$ WHICH CLAIMED TO ATTEMPT TO MOVE MONEY TO OPPOSITION/RESISTANCE FORCES IN BOTH CZECK AND AFGANISTAN REGARD HAND WEAPONS WHICH ACCORDING TO SOURCES WERE ATTEMPTING AGAINST THE WHITE HOUSE OF THE U.S.A. AND THE GOOD SERVICEMEN AND WOMEN OF THE NAVY AND THE U.S. ARMY FORCES. ALSO THE MONEY SENT TO ATTEMPT AGAINST THE U.S. ARMY FORCES AND NAVY SERVICEMEN AND WOMEN PROCURATED BOTH PETTY WEAPONS AND GROSS/NET SHIPPING REGISTRIES FOR SEASONAL RECEIPTS.

<ns.nt.hs:::library::macros::library.././Tx/forteenusesofSources//BCV::lib::ns.macros.loadspacedefMacro:::macros::library/.tx.C++.c/

<require::defMacro.Sourcesload/HTTPMedia.lib.ns.macrosources->[a]{Int}[/y/? identical ? true==BOOLEANexpression.re..//defMacro/Tx->Nolook.loadspacerN.ns> [CID/fbi/@2W->kw::May3United2.-2#3..Navy](mailto:CID/fbi/@2W-%3ekw::May3United2.-2#3..Navy)

<ns.>?

<Microspeedload->M:require::Sourcesload/HTTP:://time date connection.BCV->[list] }

##C++CODESFORLOADSPACE 2W DEFENDS KEYPASS Tx.HTTP:://[a, b, c->[Int]}}}//? true if TRUE then LOADSPACE otherwise + HTTP. is for May3.

a | p = [Bool] [Char] }} b | /y.www//..JILLBIDEN/BOOLEANexpression | a

p -> BOOL -> CHAR }} b | p (p + 1 -> a) (vs, v->[LIST] } p = 1

[Char] = p ( a | b | b + a -> [Int] -> [Int] -> [Int] + p -1 .. BOOL y/? true ?? a && b if LOADSPACE otherwise + HTTP:://Ma\* CODESPACEtrue for

Int (a, b, c, d, e) for lib::

Int(a=HTTP//Ma\*b String[a]-1@CODESPACEtrue

b -> ?? -> input::integerNameString = [ y/? true==input=false a && b

c == a | p for parser connection.save.connection.flush ( a + p + b + c)))

return::Int::varSequence:::BOOL::[String[a][-1@CODESPACE==false.connection.flush](mailto:-1@CODESPACE==false.connection.flush) )) reconnection reconnection.connect=false for input if and only if y/?trueMa\*b String[a]@return[Int].n

return Ma\*b for [a]

}

}}}

return:Int::contains:: [7, 7, 1, 4, 4, 4\*] list: sort /y

merge: Int connection.false.connection.save let(return==+.C++.CL)) a | /y? YES for sortlist::C++arrayHTTP/NameString=[connection.save.flush]-Stringvarloadspace:::bazooka.false a && b//Ma\*b String 3</5 </ 0 for sort: (-n)+3

returnString::for [a] for [BOOL]==true connection.merge.String[a:NameString]:::save.connection.flush ( C++ CL)<n> ?

defMacrospadf let(wx7878 time date –a.Tx(window)./../-a for list.false.connection.sort.YES=flush.close.save.connection.close.connection.flush }

<ns.close::lib<n>>? ####CLOSES LIBRARY FOR Tx.Window.

p | returnString::for[a] (vs, v) || Ma\*b <- [Int] <- [Int] <- {Int]

merge::sort /y <- [], []../..<n>