

dialog_pickfile

- funkce, interaktivně vybere soubor za pomoci nativního grafického dialogu (platform-dependent)
- /directory – namísto souboru vybírá adresář
- filter = '*.txt' – dává na výběr jen soubory určitého typu
- path = 'c:\temp' – počáteční cesta pro výběr souborů
- /multiple_files – umožní výběr několika souborů (vrátí pole stringů)
- /read, /write – jen změní název výběrového okna
- /must_exist – umožní výběr jen existujících souborů (aneb neumožní zadat neexistující soubor)
- /overwrite_prompt – má efekt jen při současném užití „/write“, zeptá se na potvrzení při zadání existujícího (aneb zda skutečně chceme přepsat)
- title='navez' – název dialogového okna

file_search

- vrátí seznam souborů splňujících danou specifikaci
- 1. možnost volání: 1 parameter (path) – soubory splňující danou specifikaci
- 2. možnost volání: 2 parametry (adresář + maska) – rekurzivně prohledává i podadresáře
- relativní vs. absolutní cesta
- pro relativní cestu vrací relativní i u nalezených souborů, jde změnit: /fully_qualify_path
- count=pocet – počet nalezených souborů
- x = file_search('c:\temp','*.*')

cd

- změni aktuální adresář (opět relativní vs. absolutní cesta)

pushd, popd, printd

- pushd: změni adresář na zadaný, ten současný (tj. nyní už starý) uloží do zásobníku
- popd: změni adresář na adresář z vrcholu zásobníku
- printd: vytiskne zásobník

ascii_template, read_ascii

openr, openw, openu

- otevře soubor pro vstup a/nebo pro výstup
- [unit, filename, [/get_lun]]
- /swap_if_big_endian, /swap_if_little_endian
- vsuvka – swap_endian, swap_endian_inplace

close, free_lun

- zavře (close), resp. dealokuje (free_lun) jednotku

print, printf, string

read, readf, reads

- čtení textových dat, liší se tím, odkud čte
- read: čte ze standardního vstupu; keyword: prompt='Prompt při zadávání'
- readf: čte ze souboru; 1. parametr: logical unit
- reads: čte ze stringu; 1. parametr: string
- neformátované čtení vs. formátované: format='(format codes)'

- format codes: [n] FC [+] [-] [width]
- n: kolikrát opakovat
- FC: format code
- +: normálně mají záporná čísla -, kladná nic; s tímhle mají kladná +
- -: left-justified namísto right-justified
- width: kolik znaků výstupu; pokud jich je méně, doplní se mezerama; pokud chceme nuly místo mezer, tak třeba: format='(I08)' místo format='(I8)'; pokud čteme čísla rozdílných délek, můžeme dát width=0, třeba: format='(3F0)' načte 3 reálná čísla oddělená mezerama
- FC: [width].[number of positions after decimal point]
 - A ... znaky
 - F, E, G ... reálná čísla; F: běžný zápis s desetinnou tečkou, E: zápis s exponentem, G: vybere „lepší“ z obou zápisů
 - I (B, O, Z) ... celá čísla; B, O, Z: dvojková, osmičková, šestnáctková

fstat

- informace o souboru odpovídajícím dané jednotce
- .size – velikost souboru v bytech

eof

- syntaxe: result = eof(unit)
- pokud jsme na konci souboru, vrátí 1 (true), jinak 0 (false)

strlen

strmid (expression, first, [length])

strpos (v čem, co, [pos], [/reverse_search])

strtrim (string, [flag]) nic/0 – trailing, 1 – leading, 2 – both

strsplit (string, [pattern], [/regex], length=, count=, [/extract])

strjoin (string, [delimiter])

Příklad:

```
str = 'Out, damned spot! Out I say!'
print, (STRJOIN(STRSPLIT(str, /EXTRACT), ':'))
```

regular expressions (viz stregex.pdf)

stregex (string, regexexpression, length=, [/extract])

on_ioerror, label

- návěstí, kam se má skočit při I/O problému (normálně vytiskne chybovou hlášku a zastaví program)
- pokud k chybě dojde ve stejné proceduře, co byla (už) zavolána on_io_error, pokračuje se na dané návěstí
- číslo chyby je v proměnné !error_state, text chybové hlášky v proměnné !error_state.msg

readu, writeu

- čtení/zápis binárních dat
- 1. parametr: odkud/kam, tj. logical unit
- 2. parametr (případně všechny další): proměnná/proměnné pro čtení/zápis
- co se čte je dáno typem datové proměnné (proměnných), tj. musí dopředu existovat (na rozdíl od čtení textových dat)

point_lun

- nastaví nebo naopak načte pozici ukazatele v daném souboru
- syntaxe: point_lun, unit, position
- unit > 0: nastaví ukazatel v daném souboru na position (v bytech)
- unit < 0: uloží do position aktuální polohu v daném souboru