

Systemy Operacyjne Laboratorium Nr 1

Pliki wsadowe

System DOS, a właściwie interpretator poleceń COMMAND.COM umożliwia automatyczne wykonywanie ciągu dowolnych wewnętrznych i zewnętrznych poleceń systemu, umieszczonych w pliku o dowolnej nazwie podstawowej i rozszerzeniu BAT. Interpretator poleceń wykonuje wbudowane polecenia przeznaczone do pisania programów wsadowych - :*label*, CALL, ECHO, FOR, GOTO, IF, SHIFT, PAUSE i REM oraz interpretuje parametry formalne programu wsadowego. Umożliwia to sterowanie wykonaniem ciągu poleceń zawartych w programie wsadowym, zwiększając możliwość stosowania plików wsadowych do automatyzacji działania i parametryzacji systemu.

Plik wsadowy (ang. *batch*) można utworzyć posługując się dowolnym edytorem. W pliku (programie) wsadowym mogą znaleźć się parametry formalne, oznaczane przez %0 - %9. W czasie wykonywania programu wsadowego parametry formalne zostaną zastąpione parametrami aktualnymi, podanymi w wierszu polecenia, przy czym parametrowi formalnemu %0 odpowiada nazwa programu (pliku) wsadowego, parametrowi %1 pierwszy parametr po nazwie pliku wsadowego itd.

W programach wsadowych można korzystać z wartości nadanych symbolom (zmiennym) znajdującym się w otoczeniu (ang. *environment*) interpretatora COMMAND.COM (por. polecenie SET). Użycie symbolu (zmiennej) z otoczenia ujętego w znaki % jest równoważne wpisaniu do pliku wsadowego ciągu znaków przypisanych temu symbolowi - wartości tej zmiennej.

Na przykład po wykonaniu polecenia:

```
SET DIRA=dir a:
```

wiersz programu wsadowego:

```
%DIRA%*.txt jest równoważny
```

wierszowi:

```
DIR A:*.TXT
```

Opisana metoda korzystania z symboli (zmiennych) określonych w otoczeniu istniała również (nie udokumentowana!) w wersjach systemu wcześniejszych niż MS-DOS 3.30.

Specjalne znaczenie w systemie DOS ma plik o nazwie AUTOEXEC.BAT. Podczas inicjacji COMMAND.COM sprawdza, czy w katalogu głównym dysku, z którego są ładowane programy systemowe, znajduje się plik AUTOEXEC.BAT; jeżeli go znajduje, to polecenia w nim zawarte są automatycznie wykonywane. Pliku AUTOEXEC.BAT można zatem użyć do automatycznej parametryzacji systemu i automatycznej inicjacji wykonania żądanych programów.

W programach wsadowych:

- można stosować symbole < > i >> zmiany przyporządkowania standardowych strumieni informacji;
- polecenie zmiany dysku lub katalogu ma wpływ na każde dalsze polecenie;
- w pliku wsadowym można wywołać (CALL) inne pliki wsadowe (od wersji 3.30) systemu; w wersjach wcześniejszych nazwę innego pliku wsadowego można umieścić tylko w ostatnim wierszu pliku;
- wykonanie poleceń programu wsadowego można przerwać klawiszem CTRL-BREAK.

Polecenia plików wsadowych

CALL

Wywołanie z pliku wsadowego kolejnego pliku wsadowego.

CALL *file* [*parameters*]

Znaczenie parametrów

file

Plik zawierający wywoływany program wsadowy.

parameters

Parametry aktualne programu wsadowego *file*.

Po zakończeniu wykonywania poleceń zawartych w pliku *file* są interpretowane dalsze polecenia umieszczone w danym pliku wsadowym (po CALL).

ECHO

Włączenie lub wyłączenie wyświetlania na ekranie poleceń pliku wsadowego w trakcie ich wykonywania.

[@]ECHO [ON | OFF | *message*]

Znaczenie parametrów

@

Polecenie pliku wsadowego poprzedzone znakiem @ nie jest wyświetlane na ekranie, niezależnie od wcześniej użytych poleceń ECHO (wersja 3.30 i następne).

ON | OFF | *message*

Włączenie (ON) lub wyłączenie wyświetlania na ekranie poleceń pliku wsadowego w trakcie ich wykonywania. Wartością domyślną jest ON. Tekst *message* jest wyświetlany nawet wówczas, gdy wcześniej w pliku umieszczono polecenie ECHO OFF.

FOR

Polecenie iteracji.

FOR %%v IN (*set*) DO *singlecommand*

Znaczenie parametrów

v

Jednoznakowa nazwa zmiennej; aby uniknąć pomyłek (parametry formalne programu wsadowego), nie należy stosować cyfr 0 - 9.

set ::= {*files* | *string*}

Lista składająca się z niejednoznacznych nazw plików lub innych łańcuchów znaków, w której kolejne elementy są oddzielone spacją lub innym separatorem.

singlecommand

Polecenie systemu DOS. W parametrach polecenia może wystąpić zmienna %%v.

Wykonanie polecenia polega na przypisywaniu zmiennej %%v kolejnych elementów listy *set* i iteracyjnym wykonywaniu polecenia *singlecommand*. Jeżeli w *singlecommand* znajdzie się zmienna %%v, to zostaje ona zastąpiona jej aktualną wartością. Jeśli *set* zawiera niejednoznaczną nazwę pliku *files* — ze znakami globalnymi ? lub * — to przed każdą iteracją

nazwy niejednoznaczne są rozwijane na wszystkie odpowiadające im nazwy jednoznaczne. Polecenie FOR nie może być zagnieźdżane.

Polecenie FOR może być także wykonane bezpośrednio z wiersza polecenia; w takim przypadku nazwa zmiennej powinna być poprzedzona tylko jednym znakiem %:

```
FOR %v IN (set) DO singlecommand
```

Przykład

```
FOR %%k IN (*.TXT *.DOC) DO TYPE %%k
```

Na monitorze zostanie wyświetlona zawartość wszystkich plików znajdujących się w katalogu roboczym, mających rozszerzenie TXT lub DOC.

GOTO

Polecenie skoku.

```
GOTO label
```

Znaczenie parametrów

label

Identyfikator etykiety.

Polecenie GOTO powoduje skok do wiersza pliku wsadowego zawierającego etykietę *:label* (poprzedzoną dwukropkiem). Etykietę należy umieścić w odrębnym wierszu; wiersze pliku wsadowego zaczynające się od znaku dwukropka są podczas wykonania ignorowane (tzn. są traktowane jako zawierające wyłącznie etykietę).

IF

Polecenie wykonania warunkowego.

```
IF [NOT] condition singlecommand
```

Znaczenie parametrów

condition ::= ERRORLEVEL number

Warunek przyjmie wartość TRUE, jeżeli ostatnie wykonane przed IF polecenie zwróci kod powrotu większy lub równy liczbie *number* (przyjmuje się, że kod powrotu po poprawnym wykonaniu jest równy zero).

condition ::= string1==string2

Warunek przyjmie wartość TRUE, jeżeli ciągi znaków *string1* i *string2* są identyczne. W ciągu *string* może wystąpić zmienna znajdująca się w otoczeniu w formacie *%name%*; ciąg znaków *%name%* przed wyznaczeniem wartości warunku jest zamieniany na wartość tej zmiennej otoczenia.

condition ::= EXIST file

Warunek przyjmie wartość TRUE, jeżeli plik o nazwie *file* istnieje.

Wykonanie polecenia IF polega na wyznaczeniu wartości warunku *condition* i wykonaniu polecenia *singlecommand*, jeżeli wartość ta jest równa TRUE (przy opcji NOT - FALSE).

Przykład

Jeżeli wykonamy polecenie:

```
SET TMP=f: a w pliku wsadowym znajduje się
```

polecenie:

IF %TMP%==f: DIR to wykonanie programu wsadowego spowoduje wykonanie polecenia DIR.

PAUSE

Chwilowe zatrzymanie wykonywania poleceń programu wsadowego.

PAUSE [*comment*]

Znaczenie parametrów

comment

Wartość ciągu *comment* jest ignorowana. W ciągu nie powinny znajdować się znaki <, > i |.

Polecenie powoduje zatrzymanie wykonywania poleceń pliku wsadowego, wyświetlenie ciągu znaków *comment*, a następnie komunikatu: Press any key to continue ... Wznowienie wykonywania nastąpi po naciśnięciu dowolnego klawisza z wyjątkiem CTRL-BREAK lub CTRL-C, który przerwie wykonanie pliku wsadowego. Typowym przypadkiem użycia polecenia jest przerwa na zmianę dyskiety podczas np. instalowania programu na dysku twardym.

REM

Oznaczenie komentarza w pliku .BAT.

REM [*comment*]

Znaczenie parametrów

comment

Wartość ciągu *comment* jest wyświetlana na ekranie. W ciągu tym nie powinny znajdować się znaki <, > i |.

Polecenie umożliwia umieszczenie komentarza w pliku wsadowym. Wiersz zawierający polecenie REM jest wyświetlany tylko wówczas, gdy jest włączone wyświetlanie poleceń (ECHO ON).

SHIFT

Zmiana numeracji parametrów formalnych.

SHIFT

Wykonanie polecenia SHIFT powoduje przenieście parametrów formalnych (i aktualnych) programu wsadowego. Wykonanie polecenia powoduje, że parametr %1 staje się parametrem %0, parametr %2 parametrem %1 — ogólnie parametr %(<+!) staje się parametrem %n.

Przykład

Jeżeli utworzymy plik TEST.BAT o zawartości:

```
echo off
:ety
if "%1" == "" goto end
echo #1
```

```
shift
goto ety
:end
```

to po wywołaniu: test tak działa shift otrzymamy wydruk:

```
echo off
tak
działa
shift
```

Zadania

1. Przeprowadź analizę plików config.dos, autoexec.dos, config.sys oraz autoexec.bat;
2. Jakie urządzenia i ustawienia są stosowane w celu wyświetlenia polskich znaków na ekranie?
3. Jakie urządzenia i ustawienia są stosowane, aby było możliwa pisownia polskich znaków?
4. Napisać skrypt katalog.bat, który utworzy na dysku D: w bieżącym katalogu katalog o dowolnej nazwie. Za pomocą napisanego skryptu utworzyć katalog o nazwie ROBOCZY;
5. Za pomocą polecenia COPY CON stworzyć plik zadanie1.bat kopiujący wszystkie pliki z rozszerzeniem .TXT z katalogu Windows do katalogu bieżącego;
6. Zmodyfikować skrypt katalog.bat w ten sposób aby po podaniu dwu parametrów zakładał dwa różne katalogi;
7. Zmodyfikować skrypt katalog.bat w ten sposób aby po podaniu 6 parametrów zakładał 6 różnych katalogów przy czym następny ma się znajdować w katalogu poprzednim. Po założeniu wszystkich katalogów aktywnym ma być katalog główny dysku D:;
8. Stworzyć skrypt pliki.bat, który będzie kopiował po 10 plików .exe z katalogu C:\Windows\System32 do każdego z wcześniej złożonych katalogów. Po zakończeniu kopiowania aktywnym katalogiem ma być D:\;
9. Napisać skrypt usuwający całą 6 poziomą strukturę katalogów stworzoną w zadaniu 7;
10. Napisać skrypt czytający ścieżkę katalogu TMP a następnie skopiować wszystkie pliki o nazwie 8 znakowej i dowolnym rozszerzeniu z katalogu TMP do katalogu roboczy na dysku D:;