

## Pliki wsadowe powłoki Windows.

Pliki wsadowe powłoki Windows potocznie zwane BAT-ami, są to pliki tekstowe o rozszerzeniu \*.bat zawierające zestaw komend wykonywane linijka po linijce przez interpreter powłoki.

Wykonywanie pliku wsadowego jest wywoływane przez podanie nazwy pliku, nie koniecznie z rozszerzeniem, w konsoli systemu (CMD).

Wszystkie polecenia działające w konsoli systemu (linii komend) mogą być wykorzystywane w plikach BAT. Niektóre z poleceń nie mają zbyt wielkiego sensu dla samej linii komend jednak jako programy wsadowe umożliwiają rozszerzenie funkcjonalności. Przykładem takiej komendy jest *ECHO* wyświetlające na konsoli zadany tekst. Najprostszym przykładem będzie następujący plik wsadowy:

```
@ECHO OFF
```

```
ECHO Jakis tekst
```

```
@ECHO ON
```

wynikiem wykonania będzie napis na konsoli „Jakiś tekst” (bez cudzysłówów).

Poniżej zostanie opisany wybór części komend. Parametry zawarte w nawiasach kwadratowych [] są opcjonalne. Podobne informacje można uzyskać za pośrednictwem polecenia *HELP* oraz podania nazwy komendy.

Ponieważ najłatwiejszym sposobem nauki tego typu zagadnień są przykłady – po każdej porcji komend znajduje się przykład wraz z opisem ułatwiający zrozumienie.

## Kopiowanie.

Do kopiowania w powłoce Windows służy polecenie *copy* oraz *xcopy* oto ich składnia:

**COPY** [/D] [/V] [/N] [/Y | /-Y] [/Z] [/L] [/A | /B] źródło [/A | /B]

[+ źródło [/A | /B] [+ ...]] [cel [/A | /B]]

źródło      Określa pliki do skopiowania.

/A          Oznacza plik tekstowy ASCII.

/B          Oznacza plik binarny.

/D          Zezwala na odszyfrowanie tworzonego pliku docelowego.

cel         Określa docelowy katalog lub nazwę pliku dla nowych plików.

/V         Weryfikuje, czy kopiowane pliki są poprawnie zapisywane.

/N         Używa krótkich nazw plików, jeśli są one dostępne, podczas kopiowania pliku o nazwie innej postaci niż 8.3.

/Y         Wyłącza monitowanie o potwierdzenie zastąpienia istniejącego pliku docelowego.

/-Y        Włącza monitowanie o potwierdzenie zastąpienia istniejącego pliku docelowego.

/Z         Kopiuje pliki sieciowe w trybie restartowalnym.

/L         Jeśli źródło jest łączem symbolicznym, kopiuje łącze do obie docelowego zamiast rzeczywistego pliku, do którego wskazuje łącze.

Przełącznik /Y może być ustawiony w zmiennej środowiskowej COPYCMD.

Może on być następnie zastąpiony przełącznikiem /-Y w wierszu polecenia.

Stanem domyślnym jest monitowanie przy zastępowaniu plików, chyba że polecenie COPY jest wywoływane ze skryptu wsadowego.

Aby dołączyć pliki, określ jeden plik docelowy, ale wiele plików

źródłowych (używając symboli wieloznacznych lub formatu plik1+plik2+plik3).

**XCOPY** źródło [cel] [/A | /M] [/D[:data]] [/P] [/S [/E]] [/V] [/W]

[/C] [/I] [/Q] [/F] [/L] [/G] [/H] [/R] [/T] [/U]

[/K] [/N] [/O] [/X] [/Y] [/-Y] [/Z] [/B]

[/EXCLUDE:plik1[+plik2][+plik3]...]

źródło      Określa pliki do skopiowania.

cel         Określa lokalizację i/lub nazwę nowych plików.

/A         Kopiuje tylko pliki z ustawionym atrybutem archiwalnym, nie zmieniając atrybutu.

/M         Kopiuje tylko pliki z ustawionym atrybutem archiwalnym, wyłączając ten atrybut.

/D:m-d-r   Kopiuje pliki zmienione określonego dnia lub później.

Jeśli nie podano daty, kopiuje tylko pliki, w przypadku których czas wersji źródłowej jest późniejszy niż czas wersji docelowej.

/EXCLUDE:plik1[+plik2][+plik3]...

Określa listę plików zawierających ciągi. Każdy ciąg powinien być w osobnym wierszu w plikach. Jeżeli jakiś ciąg pasuje do dowolnej części ścieżki absolutnej kopiowanego pliku, plik ten nie zostanie skopiowany. Na przykład określenie ciągu postaci \obj\ lub .obj spowoduje wykluczenie odpowiednio wszystkich plików w podkatalogu obj lub wszystkich plików z rozszerzeniem obj.

- /P Monituje przed utworzeniem każdego pliku docelowego.
- /S Kopiuje katalogi i podkatalogi, z wyjątkiem pustych.
- /E Kopiuje katalogi i podkatalogi, łącznie z pustymi.  
Działa tak samo jak kombinacja przełączników /S /E.  
Przełącznika można użyć do zmodyfikowania przełączników /T.
- /V Weryfikuje rozmiar każdego nowego pliku.
- /W Monituje o naciśnięcie klawisza przed kopiowaniem.
- /C Kontynuuje kopiowanie nawet po wystąpieniu błędu.
- /I Jeśli cel nie istnieje i kopiowany jest więcej niż jeden plik, zakłada, że cel musi być katalogiem.
- /Q Nie wyświetla nazw kopiowanych plików.
- /F Kopiując, wyświetla pełne nazwy plików źródłowych i docelowych.
- /L Wyświetla pliki, które mają być skopiowane.
- /G Zezwala na kopiowanie szyfrowanych plików do miejsca docelowego, które nie obsługuje szyfrowania.
- /H Kopiuje także pliki ukryte i systemowe.
- /R Zastępuje pliki tylko do odczytu.
- /T Tworzy strukturę katalogów, ale nie kopiuje plików. Nie uwzględnia pustych katalogów i podkatalogów. Kombinacja przełączników /T /E uwzględnia puste katalogi i podkatalogi.
- /U Kopiuje tylko pliki istniejące już w miejscu docelowym.
- /K Kopiuje atrybuty. Normalnie Xcopy resetuje atrybut tylko do odczytu.
- /N Kopiuje, używając wygenerowanych krótkich nazw.
- /O Kopiuje informacje o własności pliku i liście kontroli dostępu (ACL).
- /X Kopiuje ustawienia inspekcji plików (implikuje użycie przełącznika /O).
- /Y Wyłącza monitowanie o potwierdzenie zastąpienia istniejącego pliku docelowego.
- /-Y Włącza monitowanie o potwierdzenie zastąpienia istniejącego pliku docelowego.
- /Z Kopiuje pliki w trybie umożliwiającym ponowne uruchomienie.
- /B Kopiuje łączy symboliczne zamiast obiektu docelowego łączy
- /J Kopiuje, używając niebuforowanego We/Wy. Zalecane dla bardzo dużych plików.

Przełącznik /Y można umieścić w zmiennej środowiskowej COPYCMD.  
Jego działanie można zastąpić przełącznikiem /-Y użytym w wierszu polecenia.

XCOPY umożliwia wielokrotne kopiowanie oraz kopiowanie drzewa katalogów.

### **Zakładanie, usuwanie katalogów i plików.**

Do usuwania plików służy polecenie *DEL* a do usuwania katalogów, również rekursywnie i nie pustych *RMDIR*. Do zakładania katalogów można wykorzystać *MKDIR*. Ogólnie sama konsola i jej interpreter nie posiada wbudowanego edytora tekstu jednak istnieje możliwość ominięcia tego problemu.

***DEL [/P] [/F] [/S] [/Q] [/A[:]atrybuty]] nazwy***

*nazwy*           Określa listę jednego lub więcej plików bądź folderów.  
Aby usunąć wiele plików, można użyć symboli wieloznacznych.

Jeśli podany zostanie katalog, usunięte zostaną wszystkie pliki w tym katalogu.

- /P Monituje o potwierdzenie przed usunięciem każdego pliku.
- /F Wymusza usuwanie plików z atrybutem tylko do odczytu.
- /S Usuwa określone pliki ze wszystkich podkatalogów.
- /Q Tryb cichy, nie żąda potwierdzenia usunięcia w przypadku użycia globalnego symbolu wieloznacznego.
- /A Wybiera pliki do usunięcia na podstawie atrybutów atrybuty:
  - R Pliki tylko do odczytu
  - S Pliki systemowe
  - H Pliki ukryte
  - A Pliki gotowe do archiwizacji
  - I Pliki, których zawartość nie została zindeksowana
  - L Punkty ponownej analizy
  - Prefiks oznaczający "nie"

**RMDIR [/S] [/Q] [dysk:]ścieżka**

**RD [/S] [/Q] [dysk:]ścieżka**

- /S Usuwa wraz z określonym katalogiem wszystkie katalogi i pliki w nim umieszczone. Stosuje się do usuwania drzewa katalogów.
- /Q Tryb cichy, nie żąda potwierdzenia usuwania drzewa katalogów, jeśli została użyta opcja /S

**MKDIR [dysk:]ścieżka**

**MD [dysk:]ścieżka**

Tworzenie plików

echo %DATE% > plik1.txt

zestaw tych komend powoduje powstanie pliku tekstowego plik1.txt z zapisaną aktualną datą.

Kluczowy jest tu operator przekierowania „>” który umożliwił w tym wypadku przekazanie tego co miało pojawić się na konsoli do pliku plik1.txt ponieważ lik ten nie istniał (w założeniu) system operacyjny nie miał wyjścia i musiał go stworzyć. Tą prostą sztuczką można szybko tworzyć pliki z niewielką zawartością lub dla samego szaleństwa tworzenia :).

## Przykład:

Przykład dla powyższych instrukcji składa się z 3 plików list1.bat, list1b.bat i list1c.bat. Każdy z nich wykonuje podobną operację z niewielką modyfikacją.

Na początek list1b.bat:

@ECHO OFF

mkdir .\testowy

copy plik1.txt .\testowy

copy plik1.txt .\testowy\innanazwa.txt

copy plik1.txt .\kopia\_plik1.txt

@ECHO ON

REM wyłącz echo – czyli wyświetlanie na konsoli

REM utwórz katalog „testowy” w bieżącym katalogu

REM skopiuj plik plik1.txt i jego zawartość do

REM katalogu .\testowy

REM skopiuj zawartość pliku plik1.txt do pliku

REM innanazwa.txt w katalogu .\testowy

REM skopiuj zawartość pliku plik1.txt do bieżącego

REM katalogu do pliku kopia\_plik1.txt

REM włącz echo konsoli

Jak widać za pomocą *copy* można nie tylko kopiować pliki między katalogami ale również zmieniać ich nazwę.

Plik list1b.bat jest bat-em sprzątającym:

```
@ECHO OFF
rmdir /s /q .\testowy      REM usuń katalog „testowy” i całą jego zawartość
del .\kopia_plik1.txt      REM usuń plik kopia_plik1.txt
@ECHO ON
```

Komenda rmdir posiada 2 przełączniki: /s – wymusza usunięcie katalogu nawet jeśli nie jest on pusty, przełącznik /q – wyłącza wyświetlanie komunikatów od w/w komendy.

Plik list1c.bat:

```
@ECHO OFF
mkdir .\testowy > NUL
copy plik1.txt .\testowy > NUL
copy plik1.txt .\testowy\innanazwa.txt > NUL
copy plik1.txt .\kopia_plik1.txt > NUL
@ECHO ON
```

Niewiele różni się od pliku list1.bat jedyną zmianą jest dodanie „>NUL” - mimo iż wyłączyliśmy echo to nadal komunikaty pojawiały się na konsoli (list1.bat) aby to zmienić można przekierować wyjście operatorem > w inne ujście np. katalog lub jak tu do NIC (nul).

Plik list1d.bat:

```
@ECHO OFF
mkdir ..\kopia_testowy
xcopy .* ..\kopia_testowy
@ECHO ON
```

W tym pliku zastosowano polecenie xcopy aby skopiować zawartość bieżącego katalogu łącznie z plikami (\* - oznacza wszystko) do katalogu kopia\_testowa znajdującego się o jedno drzewo katalogowe niżej – dlaczego tak – jeśli spróbowali byśmy skopiować całą zawartość bieżącego katalogu (w tym „kopia\_testowa”) do kopia\_testowa, wymagali byśmy kopiowania samego siebie do siebie, w końcu kopia\_testowa zawiera się w katalogu bieżącym.

## Zmienne środowiskowe.

Zmienne środowiskowe są to nazwane kontenery przechowywające ciągi tekstowe. Mówiąc niedokładnie lecz prościej są to nazwy (aliasy) dla innych napisów (ciągów tekstowych).

Zmienne środowiskowe ustawia się komenda *SET* po której następuje nazwa zmiennej, znak równości i wartość zmiennej

Proszę poniższe komendy wprowadzić na konsoli CMD i sprawdzić ich rezultat :

```
SET test=ala      REM ustawia zmienną środowiskową test na wartość „ala”
%test%            REM zwraca wartość zmiennej środowiskowej test
```

Zmienne te można wykorzystywać również w plikach wsadowych w ten sam sposób. Zmienne utworzone poleceniem SET pozostają ważne do momentu zamknięcia konsoli, po zamknięciu

konsoli, w której polecenie to zostało wywołane, zmienne przez nie utworzone są kasowane.

Istnieje możliwość pozyskania tylko części zmiennej środowiskowej przy wykorzystaniu modyfikatora ~, oto sposób jego użycia:

%test:~0,2% - komenda ta spowoduje zwrócenie (wyświetlenie) 2 znaków począwszy od pozycji 0 z zmiennej środowiskowej „test”.

Niektóre zmienne środowiskowe są ustalane przy starcie systemu i pozostają ważne aż do momentu ponownego uruchomienia, jedną z nich jest PATH zawierająca ścieżki przeszukiwania dla systemu. Są również zmienne systemowe ciągle aktualizowane jak;

DATE i TIME zawierające kolejno datę i czas systemowy. Na przykład komenda %DATE% zwróci (wyświetli) aktualną datę a komenda %TIME:~3,2% zwróci aktualną minutę w godzinie.

## Instrukcja GOTO i instrukcje warunkowe.

Instrukcja GOTO czyli idź do służy do mijania części pliku wsadowego. Jej składnia to:

GOTO *etykieta* gdzie *etykieta* jest znacznym miejscem w pliku do którego ma przejść interpreter poleceń. Na przykład:

```
echo tekst1
```

```
GOTO etykieta1
```

```
echo tekst2
```

```
echo tekst3
```

```
:etykieta1
```

```
echo tekst3
```

spowoduje wyświetlenie tylko tekst1 i tekst3 ominięty zostanie fragment wyświetlający tekst2 i tekst3.

Wiele poleceń GOTO może odnosić się do tej samej etykiety.

Instrukcja warunkowa to IF czyli jeśli jej składnia jest następująca:

```
IF warunek instrukcja
```

```
IF NOT warunek instrukcja
```

Gdzie warunek – to warunek logiczny który w pierwszym przypadku musi być spełniony a w drugim niespełniony aby wywołać instrukcje „instrukcja”

## Przykład:

Przykładem jest plik list2.bat

```
@ECHO OFF
```

```
ECHO tekst1
```

```
GOTO etykieta1          REM skocz do etykiety etykieta1
```

```
ECHO tekst2
```

```
ECHO tekst3
```

```
:etykieta1              REM etykieta oznaczająca miejsce do skoku
```

```
ECHO tekst4
```

```
IF EXIST plik1.txt GOTO istnieje  REM instrukcja warunkowa skocz jeśli plik1.txt istnieje
```

```
ECHO nie ma pliku plik1.txt
```

```
GOTO end
```

```
:istnieje
```

```
ECHO plik: plik1.txt istnieje
```

```
:end
```

```
@ECHO ON
```

## Parametry przekazywane do plików wsadowych.

/CS      Umożliwia zaznaczanie wyborów uwzględniających wielkość

liter.

Domyślnie narzędzie nie uwzględnia wielkości liter.

/T limit czasu Liczba sekund, po upływie których zostanie dokonany domyślny wybór. Akceptowane są wartości z zakresu od 0 do

9999. Jeżeli wprowadzisz cyfrę 0, domyślny wybór zostanie dokonany natychmiast.

/D wybór Określa wybór domyślny po upływie nnnn s. Znak musi znajdować się w zestawie wyborów określonych przy użyciu opcji /C. Wartość nnnn musi być określona parametrem /T.

/M tekst Określa tekst komunikatu wyświetlanego przed monitem. Jeżeli tekst nie zostanie określony, narzędzie wyświetli tylko monit.

/? Wyświetla ten komunikat pomocy.

Po zakończeniu komendy w zmiennej środowiskowej %ERRORLEVEL% będzie znajdował się indeks wyboru z listy po przełączniku /C

Przykład:

@ECHO OFF

CHOICE /C:NT /D:T /T:5 /N /M:"Czy usunac katakog jakis.txt? [T-tak N-nie]:"

IF %ERRORLEVEL% == 2 GOTO s\_tak REM wybrano opcję 2 czyli T

IF %ERRORLEVEL% == 1 GOTO s\_nie REM wybrano opcję 1 czyli N

IF %ERRORLEVEL% == 0 GOTO s\_w1 REM wystąpił błąd

IF %ERRORLEVEL% == 255 GOTO s\_w1 REM j.w.

:s\_tak

ECHO wybrano TAK

GOTO end

:s\_nie

ECHO wybrano NIE

GOTO end

:s\_w1

ECHO wystąpił błąd komendy wprowadzania

:end

@ECHO ON

## ***Zadania do wykonania.***

### **Zadanie 1.**

Napisać plik wsadowy kopiujący z katalogu wywołania wszystkie pliki z rozszerzeniem .txt do katalogu kopia\_txt. Jeśli katalog ten nie istnieje należy go stworzyć jeśli istnieje należy zapytać użytkownika o decyzję czy nadpisać katalog i pliki w nim zawarte czy też nie, jeśli nie należy przerwać wykonywanie bat-a, jeśli użytkownik nie wprowadzi odpowiedzi na w/w pytanie po 5 sekundach ma nastąpić wybranie odpowiedzi domyślnej TAK.



## **Zadanie 2.**

Napisać plik wsadowy tworzący kopię zapasową katalogu wywołania w katalogu kopii o nazwie: kopia\_RRRR-MM-DD\_HH\_mm

gdzie:

RRRR – bieżący rok

MM – bieżący miesiąc

DD – bieżący dzień

HH – bieżąca godzina

mm – bieżąca minuta

Katalog kopii ma się znajdować w miejscu podanym przez użytkownika jako parametr lub w przypadku jego braku ma to być katalog o stopień wyżej w drzewie katalogów względem pozycji bieżącej (..\). Kopiowane mają być wszystkie pliki i podkatalogi. Program ma monitorować na wypadek istnienia już katalogu o nazwie jak katalog kopii, monit ten ma mieć wartość domyślną TAK przyjmowaną po 5s.

## **Zadanie 3.**

Napisać plik wsadowy jak wyżej z tą różnicą iż katalog kopii ma zostać skompresowany za pomocą programu 7-Zip. Katalog kopii jeśli jest wykorzystywany ma zostać usunięty aby pozostało jedynie archiwum z kopią zawartości katalogu bieżącego. Należy pamiętać o zabezpieczeniu przed nadpisaniem już istniejącego archiwum, monit w tej sprawie ma mieć również wartość domyślną nadpisującą istniejące archiwum.

pomoc:

<http://www.7-zip.org/faq.html>