

Windows command – delo z datotekami

- Pomembne spremenljivke okolja
 - PATH ; spisec imenikov, ki se uporabljajo pri iskanju programov. Posamezni imeniki so ločeni z podpičji
 - PATHEXT vsebuje končnice datotek, ki jih sistem obravnava kot končnice izvajalnih datotek.
 - DATE
 - TIME
 - CD
 - ERRORLEVEL
 - RANDOM

```
SET PATH=%PATH%;C:\Andrej\bin  
SET spr s preledki=vsebina s presledki  
ECHO %spr s presledki%
```

Windows command – delo z datotekami

- V lupino vgrajeni ukazi
 - CD, CHDIR
 - DIR
- Ukazi v C:\Windows\System32
 - ATTRIB
 - SUBST

 - cd..
 - cd \
 - cd (izpiše pot, kot *pwd*)

Windows command – delo z datotekami

- Pomembni ukazi za delo z datotečnim sistemom
 - PUSHD, POPD
 - TREE (grafični prikaz)
 - MD, MKDIR (tvori tudi vmesne imenike)
 - RD, RMDIR (/Q /S) ... Quiet, Subdirs
- Datotečni sistem = več logičnih zvezkov
 - VOL [pogon:]
 - LABEL [pogon:][oznaka]

Windows command – delo z datotekami

- Logični zvezki (navidezni pogoni)
 - SUBST R: .
 - SUBST R: /D ... Delete
 - SUBST R: "C:\Documents and Settings\User\My Documents"
- DIR
 - DIR /A (izpiše vse, tudi systemske datoteke in imenike)
 - DIR /AH (izpiše samo skrite datoteke in imenike)
 - DIR /AD (izpiše samo imena podimenikov)
 - DIR /A-D (izpiše samo imena datotek, ne pa tudi imena podimenikov)
 - DIR /B (izpiše samo seznam z imeni datotek, primerno za obdelavo v skriptih)

Windows command – delo z datotekami

- DIR

- DIR /S (rekurzivno izpiše tudi informacije v vseh podimenikih)
- DIR /ON (*datoteke uredi po imenih*)
- DIR /OS (datoteke uredi po velikosti)
- DIR /OD (datoteke uredi po času)
- DIR /TA (namesto časa spremembe datoteke upošteva in izpiše čas zadnjega dostopa)
- Ukaz DIR lahko uporabimo tudi za iskanje datotek. Na primer, vse datoteke *.TXT v logičnem zvezku D: dobimo z naslednjim ukazom:

```
DIR /B/S D:*.TXT
```

Windows command – delo z datotekami

- **MOVE**

- Premikanje in preimenovanje datotek med imeniki. Naenkrat lahko premaknemo več datotek, preimenujemo pa lahko samo eno datoteko.

- **REN, RENAME**

- Samo preimenovanje datotek. Naenkrat lahko spremenimo končnico tudi več datotekam.

MOVE a.txt b.txt

REN *.txt *.doc

Windows command – delo z datotekami

- **COPY**

- Pri kopiranju lahko datotekam spremenimo imena, npr. ukaz `COPY *.TXT *.OLD` prekopira vse datoteke *.TXT v datoteke *.OLD

- **DEL, ERASE**

- če brišemo z ukazom DEL se datoteke **NE PRESTAVIJO V KOŠ AMPAK V RESNICI ZBRIŠEJO!** če zahtevamo brisanje vseh datotek v neki mapi nas računalnik pred brisanjem vpraša, ali res želimo to narediti. Zbrišejo se le datoteke v trenutnem imeniku, ne pa tudi podimeniki.

DEL /P (vedno zahteva potrditev brisanja) .. Prompt

DEL /Q (nikoli ne zahteva potrditve brisanja, tudi če brišemo vse datoteke v imeniku)

DEL /S (rekurzivno zbriše tudi vse podimenike in datoteke v njih)

Windows command – delo z datotekami

- **TYPE, MORE**

- Izpis vsebine datoteke. Pri ukazu MORE se izpis na koncu vsake strani ustavi, dokler ne pritisnemo poljubne tipke.

- **ATTRIB**

- Datotekam in mapam lahko določimo attribute in zaščito. Atributi so Readonly (R), Hidden (H), System (S) in Archive (A).

ATTRIB (izpis atributov vseh datotek v imeniku)

ATTRIB +R -H *.BAT (nastavi Readonly in zbriše Hidden vsem *.BAT datotekam)

- Najpomembnejši atribut je Readonly. Datoteke, ki imajo atribut Readonly, so zaščitene pred brisanjem, preimenovanjem, premikanjem in spreminjanjem vsebine. Lahko pa takšno datoteko prekopiramo in spreminjamo njeno kopijo.

Windows command – delo z datotekami

- CACLS

- Datotečni sistem FAT ne pozna zaščit, zato ta ukaz deluje le v datotečnem sistemu NTFS.

/T Nastavi ACLje datotekam v imeniku in podimenikih (Tree)

/E Spremeni ACL namesto da bi se zamenjal. (Edit)

/C Nadaljuj ob napakah glede pravic dostopa. (Continue)

/G user:perm Dodaj uporabniku določene pravice. (Grant)

R Read

W Write

C Change (read/write)

F Full control

/D user:perm Odvzemi uporabniku določene pravice. (Deny)

CACLS Imena.txt /E /G "Power Users":R (Dodaj zaščito "Readonly" datoteki Imena.txt - zaščita je različno od atributa Readonly)

CACLS Imena.txt /E /G "FinanceUsers":F (Postavi zaščito "Full Control permission" datoteki Imena.txt)

CACLS C:\Dokumenti /E /T /C /G "FinanceUsers":F (Postavi zaščito "Full Control permission" imeniku C:\Dokumenti in vsem podimenikom)

Windows command – delo z datotekami

- ASSOC [.ext[=[fileType]]]
 - za podano končnico izpiše oz. spremeni **ime določila**, ki določa program oziroma ukaz
- FTYPE [fileType[=[openCommandString]]]
 - za vsako določilo izpiše oz. spremeni **ukazno vrstico**, ki se bo izvedla

```
ASSOC .txt
```

```
.txt=txtfile
```

```
FTYPE txtfile
```

```
txtfile=%SystemRoot%\system32\notepad.exe %1
```

Windows command – delo z datotekami

- Ostali ukazi

- HELP
- CHKDSK pregleda disk
- CHKNTFS status logičnih naprav
- CIPHER šifriranje in dešifriranje datotek in imenikov v datotečnem sistemu NTFS
- COMPACT stiskanje particij
- NTFS EXPAND raztegovanje stisnjenih datotek
- DEFRAG defragmentiranje diska
- CONVERT pretvorba datotečnega sistema FAT v NTFS (obratno ne gre!)

Windows command – skriptni jezik

- spremenljivka CMDEXTVERSION
 - V novejših (W2k, XP, 2003 Server) sistemih ima ta spremenljivka vrednost 2.
- Vsak ukaz se najprej izpiše, šele nato izvede, razen če je pred njim @, torej:

```
@ECHO OFF
```

```
ECHO Hello World!
```

```
ECHO.
```

Windows command – skriptni jezik

- Ustvari imenik, vanj prekopira vse datoteke iz trenutnega imenika, nato ustvarjeni imenik izbriše:

```
MKDIR Trash  
COPY * Trash  
RMDIR Trash /S /Q
```

- Spremenljivke:

```
@ECHO OFF  
SET STR=Hello World  
SET STR S PRESLEDKI = Ali res razumes spremenljivke?  
ECHO %STR%  
ECHO %STR S PRESLEDKI %  
SET STR=  
SET STR S PRESLEDKI =
```

Windows command – spremenljivke

- Spremenljivke se obravnavajo kot nizi znakov. Zato obstaja nekaj operacij nad podnizi. Zelo uporabni sta naslednji operaciji:
 - %Variable:~Position,Length% - vrne del vrednosti spremenljivke Variable, vrnjeni podniz se začne od mesta Position (od 0 dalje) in ima dolžino Length
 - %Variable:Replacement=Original% - vrne vrednost spremenljivke Variable, vendar pa v njej vse podnize Replacement zamenja z nizom Original. Vrednost spr. se ne spremeni.
 - Z naslednjim stavkom preverimo, ali se vrednost spremenljivke %ODGOVOR% začne s črko Y:

```
IF /I "%ODGOVOR:~,1%" EQU "y"
```

Windows command – aritmetika

- Stavku SET dodamo kretnico /A. V tem primeru se presledki in posebni zanki obravnavajo drugače ter jih ne smemo uporabljati v imenih spremenljivk, imena spremenljivk pa pišemo brez znakov za procent

```
@ECHO OFF
SET A=5
SET /A B = 2*A-1
SET /A B = B+1
ECHO A=%A%, B=%B%
```

- Podprte so aritmetične operacije seštevanje (+), odštevanje (-), množenje (*), celoštevilčno deljenje (/) in ostanek pri deljenju (%).

Windows command – interaktivni skripti

- Naslednji ukaz izpiše na ekran "Vnesi število: " in potem počaka, da uporabnik nekaj vnese. Vnešeno besedilo se shrani v spremenljivko STEVILO.

```
SET /P STEVILO="Vnesi število "
```

- Skriptom lahko ob zagonu podamo argumente, ki se obravnavajo pozicijsko:
 - %0 je ime skripta,
 - %1 je prvi argument,
 - %2 je drugi argument, itd.

Windows command – spremenljivke

- Vrednost spremenljivke lahko uporabimo tako, da tvorimo novo ime, novo vrednost spremenljivke ali pa celo sestavimo kakšen ukaz.

```
@ECHO OFF
SET A=ENA
SET D=DIR
SET E=*.BAT
SET %A%DVA=TRI%A%
ECHO %ENADVA%
MK%D% Test
%D% %E%
```

Windows command – pogojni stavki

- Na voljo imamo tri pogojne stavke, katerih splošna oblika je naslednja:
 - IF pogoj (statements) ELSE (statements)
 - IF NOT pogoj (statements) ELSE (statements)
 - IF DEFINED variable (statements)
 - IF EXIST filename (statements)
- Stavek ELSE ni obvezen, tudi oklepaje lahko v nekaterih primerih izpustimo. Če želimo pogojni stavek napisati čez več vrst, moramo vrstico deliti takoj za okroglim oklepajem, tako da dobimo naslednjo obliko:
 - IF pogoj (
Statements
) ELSE (
Statements
)

Windows command – pogojni stavki

- Pogoj, ki ga uporabimo v stavku IF, je vedno primerjava nizov znakov. Uporabimo lahko naslednje operatorje:
 - s1 == s2 (ali sta niza enaka?)
 - s1 EQU s2 (ali sta niza enaka?)
 - s1 NEQ s2 (ali sta niza različna?)
 - s1 LSS s2 (ali je prvi niz manjši od drugega?)
 - s1 LEQ s2 (ali je prvi niz manjši ali enak od drugega?)
 - s1 GTR s2 (ali je prvi niz večji od drugega?)
 - s1 GEQ s2 (ali je prvi niz večji ali enak od drugega?)

Windows command – pogojni stavki

- Pri primerjavi nizov znakov se male in velike črke razlikujejo. Če ne želimo razlikovanja, dodamo za ukazom IF kretnico /I.

```
@ECHO OFF
IF /I "%1" == "ena" (
ECHO prvi parameter
ECHO je ena
) ELSE (
ECHO prvi parameter ni ena
)
```

Windows command – parametri

- Parametri so pozicijsko številčeni %1, %2, ... Uporaben je ukaz **shift**, ki parametre pomakne za eno pozicijo v levo (odstrani prvi parameter, drugi parameter postane prvi, tretji postane drugi, itd.)

```
@ECHO OFF
:ZANKA
IF "%1" EQU "" GOTO :KONEC
ECHO %1
SHIFT
GOTO ZANKA
:KONEC
```

Windows command – zanke

- Zanke v skriptih lahko tvorimo s stavkom FOR. Na voljo je več oblik tega ukaza.

FOR %%variable IN (fileSet) DO (statements)

FOR /R path %%variable IN (fileSet) DO (statements)

FOR /D %%variable IN (directorySet) DO (statements)

FOR /R path /D %%variable IN (directorySet) DO (statements)

FOR /L %%variable IN (init,step,end) DO (statements)

FOR /F "options" %%variable IN (fileSet) DO (statements)

- Pri vseh oblikah stavka FOR velja, da je **ime spremenljivke označene kot variable lahko dolgo le en znak!** Njeno vrednost v zanki dobimo tako, da zapišemo %%variable

Windows command – FOR

- če želimo stavek FOR napisati čez več vrstic, moramo vrstico deliti takoj za okroglim oklepajem. Primeri .
 - izpiše imena vseh datotek v trenutnem imeniku.
FOR %%N IN (*) DO (ECHO %%N)
 - izpiše vse datoteke v podanem imeniku in v vseh njegovih podimenikih.
FOR /R C:\ %%N IN (*) DO (ECHO %%N)
 - izpiše vse imenike v trenutnem imeniku.
FOR /D %%N IN (*) DO (ECHO %%N)
 - izpiše vse imenike v podanem imeniku in v vseh njegovih podimenikih.
FOR /R C:\ /D %%N IN (*) DO (ECHO %%N)

Windows command – FOR

- Primeri:

- C-jevska zanka for in izpiše števila od vključno 0 do vključno 10 s korakom 2, torej se izpišejo števila 0,2,4,6,8,10.

```
FOR /L %%N IN (0,2,10) DO (ECHO %%N)
```

- pregleduje tekstovno datoteko STUDENTS.TXT, najprej preskoči prvih 10 vrstic, nato pa v vsaki naslednji vrstici vzame prve tri besede in peto besedo, ter jih izpiše

```
FOR /F "skip=10 tokens=1-3,5" %%A IN (STUDENTS.TXT) DO (  
ECHO %%A %%B %%C %%D  
)
```


Windows command – FOR

- Stavek FOR nam omogoča elegantne operacije nad polji, kakor kaže naslednji primer. Moramo pa upoštevati, da lupina Windows command polj pravzaprav ne pozna, za njo so npr spremenljivke T[1], T[2], T[3] itd. čisto navadne spremenljivke s podobnim imenom.

```
@ECHO OFF
SET T[1]=5
SET T[2]=10
SET T[3]=15
SET T[4]=20
SET SUM=0
FOR /L %%A IN (1,1,4) DO (
SET /A SUM = SUM + T[%%A]
)
ECHO SUM=%SUM%
```

Windows command – FOR

- FOR omogoča shranjevanje rezultata poljubnega ukaza v spremenljivko. To dosežemo z uporabo kretnice /F in ključno besedo "USEBACKQ" kot opcijo. Ukaz, ki se naj izvede, mora biti podan v vzvratnih narekovajih.

```
FOR /F "USEBACKQ DELIMS== TOKENS=2" %%T IN (`ASSOC  
.GIF`) DO SET NAME=%%T  
FOR /F "USEBACKQ DELIMS== TOKENS=2" %%T IN (`FTYPE  
%NAME%`) DO SET PROG=%%T  
ECHO %PROG%
```

Windows command – skoki

- GOTO

```
@ECHO OFF
ECHO zacetek
GOTO END
ECHO sredina
:END
ECHO konec
```

Pogosto potrebujemo: GOTO :EOF (skok na konec skripta)

- CALL – klic podprograma, vrne se ko pride do konca skripta

```
@ECHO OFF
SET UKAZ=NULL
FOR /L %%A IN (1,1,5) DO (
CALL :OBDELAVA %%A
)
GOTO :EOF
:OBDELAVA
SET /P UKAZ="Vnesi ukaz: "
ECHO Pod zaporedno številko %1 si vnesel ukaz %UKAZ%.
GOTO :EOF
```

Windows command – FOR in CALL

- V zanki FOR se v vsaki iteraciji ponovno **ovrednotijo le spremenljivke, pred katerimi je %%**, torej s samo zanko povezane spremenljivke. Ostale se ovrednotijo le enkrat, ob vstopu v skript ali **podprogram**. Naslednji skript ne deluje – težave z vrednotenjem spremenljivke UKAZ (vedno bi bila prazna).

```
@ECHO OFF
SET UKAZ=NULL
FOR /L %%A IN (1,1,5) DO (
SET /P UKAZ="Vnesi ukaz: "
ECHO Pod zaporedno številko %%A si vnesel ukaz %UKAZ%.
)
```

- Skript s prejšnje prosojnice deluje, spremenljivka UKAZ se ovrednoti pravilno.

```
@ECHO OFF
SET UKAZ=NULL
FOR /L %%A IN (1,1,5) DO (
CALL :OBDELAVA %%A
)
GOTO :EOF
:OBDELAVA
SET /P UKAZ="Vnesi ukaz: "
ECHO Pod zaporedno številko %1 si vnesel ukaz %UKAZ%.
GOTO :EOF
```

Windows command – veriženje

- V isti vrstici lahko naštejemo več ukazov. To imenujemo veriženje ukazov. Operatorji pri veriženju ukazov so:
 - ukaz1 & ukaz2 (najprej izvede ukaz1 nato pa ukaz2)
 - ukaz1 && ukaz2 (najprej izvede ukaz1 in če **NE pride** do napake nato izvede ukaz2)
 - ukaz1 || ukaz2 (najprej izvede ukaz1 in če **pride** do napake nato izvede ukaz2)
- Z okroglimi oklepaji lahko določimo vrstni red, v katerem se bodo ukazi izvajali.

```
DIR *.LOG || ECHO Nisem našel nobene datoteke *.LOG
```

```
IF EXIST C:\Vaje\Naloga.txt DEL C:\Vaje\Naloga.txt & ECHO Zbrisano
```

```
CD C:\VSO && (COPY *.TXT E:\ & COPY *.BAK E:\) && (DEL *.TXT & DEL *.BAK)
```

Windows command – ostali ukazi

- CLS zbriše ekran
- COLOR nastavi barvo ozadja in barvo teksta,
- DOSKEY - prikroji mehanizem hranjenja zgodovine ukazov
- SETLOCAL shrani trenutne vrednosti vseh spremenljivk
- ENDLOCAL povrne shranjene vrednosti spremenljivk
- EXIT [/B] [exitCode] zaključi skript in zapre okno ali ne (/B)
- PAUSE ustavi izvajanje skripta, dokler uporabnik ne pritisne poljubne tipke
- REM vse do konca vrstice je komentar
- PROMPT - spremeni pozivni niz
- TITLE spremeni tekst, ki je napisan zgoraj v imenu okna
- VER izpiše, katero različico operacijskega sistema uporabljamo

Windows command – upravljanje OS

- DATE nastavi datum
- DATE /T izpiše datum
- TIME nastavi čas
- TIME /T izpiše čas
- SHUTDOWN
 - L logoff
 - S shutdown (administrator)
 - R restart (administrator)
 - C comment
 - T xx delay xx seconds

SHUTDOWN -R -C "Zaustavitev sistema čez 5 minut!" -F -T 300

Windows command – NET USER

- NET USER - tvorjenje, spreminjanje in brisanje uporabnikov
- NET ACCOUNTS - spreminjanje parametrov v zvezi z gesli posameznih uporabnikov
 - Dodajanje uporabnika:
NET USER username [password] /ADD
 - Brisanje uporabnika:
NET USER username /DELETE
 - Spreminjanje gesla uporabnika:
NET USER username password

Windows command – NET ACCOUNTS

- Parametre v zvezi z gesli posameznih uporabnikov nastavimo z ukazom NET ACCOUNTS.
 - Nastavitev, da morajo gesla vsebovati vsaj šest znakov in da potečejo vsakih 7 dni:
NET ACCOUNTS /MINPWLEN:6 /MAXPWAGE:7
 - Nastavitev, da so lahko gesla poljubno kratka in da nikoli ne pretečejo:
NET ACCOUNTS /MINPWLEN:0 /MAXPWAGE:UNLIMITED
- Ukaza NET USER in NET ACCOUNTS brez parametrov lahko izvedejo vsi uporabniki, pri čemer se le izpišejo podatki o tem, kateri uporabniki so na sistemu in kakšna je politika v zvezi z gesli.

Windows command – Procesi (TASKLIST)

- Porabo sistemskih sredstev nadzorujemo z ukazom TASKLIST. Če ne podamo nobenih kretnic se izpiše seznam vseh procesov. Uporabne so npr. naslednje kombinacije kretnic:
 - Izpiše podrobnosti o procesih:
TASKLIST /V
 - Izpiše vse procese določenega uporabnika:
TASKLIST /FI "USERNAME eq uporabnik"
 - Izpiše vse procese in za vsakega vse storitve:
TASKLIST /SVC
 - Izpiše vse procese in za vsakega vse knjižnice DLL, ki jih ta proces uporablja (dolga izpis!):
TASKLIST /M
 - Izpiše vse procese, ki uporabljajo knjižnice DLL, katerih ime ustreza vzorcu GDI*, npr. GDI32.DLL:
TASKLIST /M GDI*

Windows command – Procesi (TASKKILL)

- TASKKILL. Procese lahko ukinemo na prijazen način (enako ukazu exit v programu) ali pa nasilno. Naenkrat lahko ubijemo več procesov. Običajno je, da ob ubijanju procesa zahtevamo tudi, da se ubijejo vsi njegovi potomci.
 - Ubije proces notepad.exe (IMage)
TASKKILL /IM notepad.exe
 - Ubije proces notepad.exe na nasilen (Forcefully) način:
TASKKILL /F /IM notepad.exe
 - Ubije procesa s številko PID 1001 in 1003 ter vse njune potomce (Tree):
TASKKILL /T /PID 1001 /PID 1003

Windows command – Poganjanje z RUNAS

- Z RUNAS poženemo proces v imenu drugega uporabnika. Uporabno za manjša administratorska opravila ali preizkus skriptov, če jih bo določen uporabnik lahko pognal.
 - Sprememba gesla, ne da bi se prijavili kot administrator (seveda bo ukaz deloval le, če bomo na poziv vnesli pravilno geslo za administratorja)
RUNAS /USER:Admin "NET USER uporabnik novo_geslo"
 - Zbrišemo geslo za uporabnika uporabnik:
RUNAS /USER:Admin "NET USER uporabnik \"\""
 - Sprožimo postopek spreminjanja gesla za uporabnik, računalnik bo vprašal za novo geslo:
RUNAS /USER:Admin "NET USER uporabnik *"

Windows command – AT

- AT uporabljamo za poganjanje opravil ob določenem času, enkrat ali ciklično. Primeri:
 - ob 24:00 poženi Defrager.bat
AT 24:00 Defrager.bat
 - ob 6:00 vsak torek (Tuesday) poženi DiskClean.bat
AT 6:00 /EVERY:T DiskClean.bat
 - ob 7:00 vsak ponedeljek (Monday), sredo (Wednesday) in petek (Friday)...
AT 7:00 /EVERY:M,W,F MapNtwkDrive.bat 2.
 - prekliči vsa čakajoča opravila
AT /DELETE /YES