

Lihtne skriptifail

@ Echo off

REM See on esimene lihtne skriptifail, mis salvestab argumendina

REM etteantud kataloogis olevate

REM failide nimekirja argumendiga etteantud faili.

Dir %1 > %2

@echo off – vaikimisi kuvatakse kõik skriptifailis olevad käsud command prompt aknas, „echo off“ keelab sellise tegevuse. @ märk tähendab, et ei kuvata ka „echo off“ käsku. Skriptifailide sellise käsuga alustamine on hea tava.

REM – selle käsuga algavat rida käsuinterpretaator ei loe – kasutatakse kommentaaride kirjutamiseks

dir – kuvab etteantud kataloogi sisu. > suunab dir käsu väljundi etteantud faili. %1 ja %2 on vastavalt esimene ja teine argument, millega skriptifail käivitatakse. Argumente saab olla kuni üheksa.

Seega tuleb ülaltoodud skriptifaili käivitada järgmiselt:

„skriptifail.bat kataloog faililisting.txt“

SET käsk ja kasutaja sisestuse küsimine

Kasutaja sisestuse küsimine skripti täitmise ajal käib käsu „SET“ abil. SET käsk on muutujate väärtustamiseks. Näiteks:

SET muutuja=12

Muutujale nimega „muutuja“ omistatakse väärtus 12. Edasi saaks „muutuja“ väärtust kasutada. Näitkes:

echo %muutuja%

Kuvab command prompt aknasse väärtuse 12.

SET käsul on kaks võtit /a ja /p.

/a võti tähendab, et muutujale omistatakse arvuline väärtus, mis arvutatakse võrdusmärgile järgnevast avaldisest. Näiteks:

SET /a muutuja=2+3*5

Sellisel juhul saab muutuja väärtuseks 17 (järgitakse tehete järjekorda).

/p võti võimaldab muutuja väärtuse küsida kasutajalt. Näiteks:

SET /p muutuja=Sisesta oma nimi:

Selline käsk kuvab command prompt aknasse rea „Sisesta oma nimi:“ ja jääb ootama kasutaja sisestust, mille lõpetab ENTER klahvi vajutamine.

Valiklause ja GOTO

Valiklauseks on klassikaline IF lause. Süntaks: IF [tingimus] (käsk1) ELSE (käsk2)

Näiteks:

IF %a% == %b% (ECHO a==b) ELSE (ECHO a!=b)

Tingimuse võrdlusoperaatorid on:

EQU – võrdub

NEQ – ei võrdu

LSS – on väiksem kui

LEQ – on väiksem või võrdne kui

GTR – on suurem kui

GEQ – on suure, või võrdne kui

Veel saab tingimuseks kasutada:

„**defined**“ ja „**not defined**“ - testitakse kas muutuja on defineeritud

„**exist**“ ja „**not exist**“ - näiteks faili või kataloogi olemasolu kontrollimiseks.

„**errorlevel**“ - errorlevel on eriline muutuja, kus hoitakse viimase käsu lõpukoodi (mis käsu eduka täitmise korral on 0), kui käsku edukalt ei täidetud, siis on see midagi muud olenevalt käsust ja selle täitmisel tekkinud veast.

Sageli kasutatakse koos IF lausega ka GOTO käsku. GOTO käsk võimaldab osa koodiridu vahele jätta ja „hüpata“ kohe vajaliku koodiosa juurde. Näiteks:

```
IF defined a goto :olemas
set a=vaikimisi v22rtus
:olemas
echo %a%
```

Tsükkel

Korduvate tegevuste sooritamiseks on olemas FOR lause, mille süntaks on:

for %%muutuja in (kogumik) do (käsud)

Muutuja väärtustatakse järjest kõigi kogumikku kuuluvate objektide või väärtusega.

Vaikimisi eeldatakse, et kogumikku kuuluvad failid või stringid. Näiteks:

For %%x in (*) do (echo %%x)

For %%x in („üks“, „kaks“, „kolm“) do (echo %%x)

Esimene käsk kirjutab command prompt aknasse kõik aktiivses kataloogis olevad failinimed.

Teine käsk kirjutab command prompt aknasse stringid „üks“, „kaks“, „kolm“

Sama käsk kataloogide nimede saamiseks nõuab võtit /d

For /d %%x in (*) do (echo %%x)

Kui soovimes saada failide nimekirja ka etteantud kataloogi alamkataloogides olevatest failidest, siis tuleb kasutada võtit /r

For /r c:\windows %%x in (*) do (echo %%x)

Kui soovime, et kogumik ei sisaldaks mitte faile või katalooge vaid hoopis mingit väärtuste seeriat (mis on programmeerijatele harjumuspärasem), siis kasutame võtit /l
For /l %%muutuja in (algus, samm, lõpp) do (käsud)

Näiteks:

for /l %%x in (1,1,10) do (echo %%x)

X väärtustatakse arvudega 1 kuni 10 sammuga 1.

Harjutused

- 1) Looge skriptifail, mis võimaldab argumentina sisse anda nime. Kui nime ei antud, siis küsitakse seda kasutaja käest skripti täitmise ajal. Lõpuks kasutajat tervitatakse nimepidi.
- 2) Looge skriptifail, mis lihtsustaks xcopy käsu kasutamist. Eesmärgiks on, et esimese argumentina antud kataloogi sisu kopeerida koos alamkataloogide ja peidetud failidega teise argumentiga antud asukohta. NB! Vaikimisi on olemas selline silt nagu :eof
- 3) Looge skriptifail, mis võimaldab kasutajal argumentidena sisestada kolmnurga külgede pikkused ja arvutab kolmnurga übermõõdu. Kui kasutaja sisestab vale arvu argumente, siis teavitatakse teda sellest. Samuti kontrollitakse, kas selline kolmnurk saab olemas olla (iga kahe külje summa peab olema suurem kui kolmas külge).
- 4) Looge skriptifail, mis leiab esimese argumentiga etteantud kataloogist kõik .mp3 laiendiga failid ja kopeerib need teise argumentiga etteantud kataloogi uue nimekujuga „muusika-[vanafailinimi]“. Argumentide olemasolu kontrollitakse.
- 5) Looge skriptifail, mis kirjutab kasutaja antud argumentidega määratud faili määratud ulatuses korrutustabeli.

Iseseisev ülesanne nr. 4

Koostage skriptifail vabalt valitud tegevuse (võiks olla mõni vajalik tegevus) läbiviimiseks. Skriptifail peab sisaldama vähemalt ühte valiklauset ja for tsüklit. Lisaks veel vähemalt 3 käsku.