

# Лабораторная 3

## 1-3. Знакомство с PowerShell

4. Вывести содержимое каталога Windows по указанному в табл. формату на экран и в текстовый файл.

|      |              |            |                |
|------|--------------|------------|----------------|
| 1, 6 | Только файлы | По размеру | Размер > 10000 |
|------|--------------|------------|----------------|

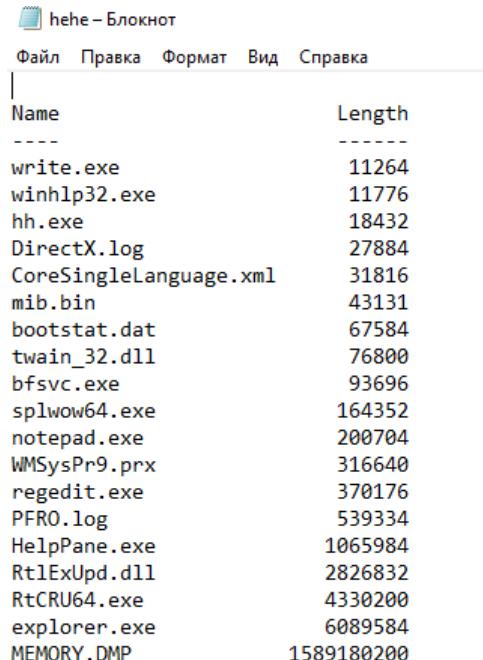
Вывод на экран:

```
Get-ChildItem -Path "C:\Windows" -File | Where-Object {$_.Length -gt 10000} | Sort-Object Length | Select-Object Name, Length
```

```
PS C:\WINDOWS\system32> Get-ChildItem -Path "C:\Windows" -File | Where-Object {$_.Length -gt 10000} | Sort-Object Length | Select-Object Name, Length
Name          Length
----          -----
write.exe      11264
winhlp32.exe   11776
hh.exe        18432
DirectX.log    27884
CoreSingleLanguage.xml 31816
mib.bin       43131
bootstat.dat  67584
twain_32.dll  76800
bfsvc.exe     93696
splwow64.exe  164352
notepad.exe   200704
WMSysPr9.prx  316640
regedit.exe   370176
PFRO.log     539334
HelpPane.exe  1065984
RtlExUpd.dll  2826832
RtCRU64.exe   4330200
explorer.exe  6089584
MEMORY.DMP    1589180200
```

Вывод в текстовый файл:

```
Get-ChildItem -Path "C:\Windows" -File | Where-Object {$_.Length -gt 10000} | Sort-Object Length | Select-Object Name, Length | tee C:\уник\lab2\hehe.txt
```



| Name                   | Length     |
|------------------------|------------|
| ---                    | -----      |
| write.exe              | 11264      |
| winhlp32.exe           | 11776      |
| hh.exe                 | 18432      |
| DirectX.log            | 27884      |
| CoreSingleLanguage.xml | 31816      |
| mib.bin                | 43131      |
| bootstat.dat           | 67584      |
| twain_32.dll           | 76800      |
| bfsvc.exe              | 93696      |
| splwow64.exe           | 164352     |
| notepad.exe            | 200704     |
| WMSysPr9.prx           | 316640     |
| regedit.exe            | 370176     |
| PFRO.log               | 539334     |
| HelpPane.exe           | 1065984    |
| RtlExUpd.dll           | 2826832    |
| RtCRU64.exe            | 4330200    |
| explorer.exe           | 6089584    |
| MEMORY.DMP             | 1589180200 |

5. Вывести в текстовый файл список свойств процесса, возвращаемый командлетом Get-Process и на экран – их общее количество.

```
PS C:\уник\lab2> $process = get-process
PS C:\уник\lab2> $process>5.txt
PS C:\уник\lab2> $process.count
230
PS C:\уник\lab2> type 5.txt
```

| Handles | NPM(K) | PM(K) | WS(K) | CPU(s) | Id    | SI | ProcessName        |
|---------|--------|-------|-------|--------|-------|----|--------------------|
| 627     | 31     | 63932 | 5420  | 11,73  | 10916 | 1  | ACCStd             |
| 562     | 57     | 26756 | 3448  | 6,02   | 9540  | 1  | AcerPortal         |
| 222     | 9      | 3752  | 6484  | 0,06   | 4092  | 0  | AdminService       |
| 561     | 25     | 17068 | 29092 | 5,02   | 8440  | 1  | AdobeCollabSync    |
| 466     | 19     | 8052  | 18372 | 0,38   | 9648  | 1  | AdobeCollabSync    |
| 171     | 10     | 1444  | 6972  | 0,08   | 4144  | 0  | AdobeUpdateService |
| 223     | 15     | 3440  | 14996 | 0,75   | 4336  | 0  | aesm_service       |
| 151     | 8      | 2720  | 8396  | 9,36   | 2280  | 0  | AggregatorHost     |
| 284     | 17     | 8192  | 17948 | 2,41   | 4172  | 0  | AGMService         |
| 223     | 14     | 22508 | 20396 | 0,44   | 13940 | 1  | ai                 |
| 223     | 15     | 48140 | 35820 | 0,75   | 14252 | 1  | ai                 |

6. Создать текстовый файл, содержащий список выполняемых процессов, упорядоченный по возрастанию указанного в табл.6 параметра.

| Номера бригад | Список выводимых параметров процессов | Сортировать по значению параметра | Вывести процессы, у которых |
|---------------|---------------------------------------|-----------------------------------|-----------------------------|
| 1, 4          | Имя процесса, BasePriority, Company   | Имя процесса                      | BasePriority > 7            |

```
Get-process | Where-Object {$_._basepriority -gt 7} | Sort-Object Name | Select-Object Name, basepriority, company | Out-File -FilePath "C:\services.txt"
```

7. Создать HTML-файл, содержащий список выполняемых процессов, упорядоченный по возрастанию BasePriority.

```
Get-process | Where-Object {$_.basepriority} | Sort-Object basepriority | Select-Object Name, PS  
C:\уник\lab2> pany | convertto-html > processes.html
```

8. Найти суммарный объем всех графических файлов ( **bmp, jpg**), находящихся в каталоге.

```
PS C:\yume\lab2> $size = 0
PS C:\yume\lab2> foreach ($f in (Get-ChildItem C:\yume -Recurse | Where-Object {$_.Extension -eq ".mp3" -or $_.Extension -eq ".m4a"})) { $size += $f.Length }
PS C:\yume\lab2> $size
14595819
PS C:\yume\lab2>
```

9. Вывести на экран сведения о ЦП компьютера.

```
PS C:\уник\lab2> Get-WmiObject Win32_Processor | format-list

Caption          : Intel64 Family 6 Model 78 Stepping 3
DeviceID         : CPU0
Manufacturer     : GenuineIntel
MaxClockSpeed    : 2601
Name             : Intel(R) Core(TM) i7-6500U CPU @ 2.50GHz
SocketDesignation : U3E1
```

10. Найти максимальное, минимальное и среднее значение времени выполнение командлетов dir и ps

Для командлетов dir:

```
PS C:\уник\lab2> $times = 1..10 | foreach-object {{measure-command {dir}}.totalseconds}
PS C:\уник\lab2> $times
0,0095254
0,0011266
0,0011011
0,0011192
0,0010673
0,063553
0,0007189
0,0006733
0,0006648
0,00066
PS C:\уник\lab2> $times | measure-object -maximum -minimum -average

Count      : 10
Average    : 0,00802096
Sum        :
Maximum   : 0,063553
Minimum   : 0,00066
Property  :
```

Для командлетов ps:

```
PS C:\уник\lab2> $times = 1..10 | foreach-object {{measure-command {ps}}.totalseconds}
PS C:\уник\lab2> $times
0,0097389
0,0056649
0,0033079
0,0030787
0,0031919
0,0032326
0,0028665
0,0033934
0,0028305
0,0027632
PS C:\уник\lab2> $times | measure-object -maximum -minimum -average

Count      : 10
Average    : 0,00400685
Sum        :
Maximum   : 0,0097389
Minimum   : 0,0027632
Property  :
```

11. Разработать командлет для:

1. нахождения количества чисел, хранящихся в файле nn.txt
  2. нахождения количества наибольших чисел, хранящихся в файле nn.txt
1. Создаем функцию для подсчета количества чисел в файле

```
PS C:\уник\lab2> function number_count{
>> param([string]$filepath)
>> $total_number=0
>> $num = get-content $filepath | foreach-object {$total_number +=1}
>> return $total_number
>> }
```

```
PS C:\уник> type nn.txt
40
35
21
84
63
93
80
85
75
77
50
17
35
7
6
53
37
PS C:\уник> number_count nn.txt
17
PS C:\уник>
```

2. создаем функцию для нахождения количества наибольших чисел

```
PS C:\уник> function Get-MaxNumberCount {
>>   param([string]$filepath)
>>   $maxNumber = $null
>>   $maxCount = 0
>>   $numbers = Get-Content $filepath | ForEach-Object {
>>     if ($_.match '^-\?\d+$') {
>>       $currentNumber = [int]$_
>>       if ($maxNumber -eq $null -or $currentNumber -gt $maxNumber) {
>>         $maxNumber = $currentNumber
>>         $maxCount = 1
>>       } elseif ($currentNumber -eq $maxNumber) {
>>         $maxCount++
>>       }
>>     }
>>   }
>>   return @{
>>     MaxNumber = $maxNumber
>>     Count = $maxCount
>>   }
PS C:\уник> get-maxnumbercount nn.txt
Name                Value
----                -----
Count               1
MaxNumber          93
```