

## 4.1 бригада

Получаем только файлы (не папки), размер > 10000 байт, сортируем по размеру ->  
Записываем тот же результат в текстовый файл

```
PS C:\Windows> Set-Location C:\Windows
PS C:\Windows>
PS C:\Windows> Get-ChildItem -File |
>> Where-Object { $_.Length -gt 10000 } |
>> Sort-Object Length |
>> Format-Table Name, Length, LastWriteTime
Name          Length LastWriteTime
----          ----  -----
winhlp32.exe    12288 6/7/2025 6:34:33 PM
Professional.xml 12481 4/1/2024 11:43:05 AM
hh.exe         23040 6/7/2025 6:33:57 PM
mib.bin        43131 4/1/2024 11:43:18 AM
bootstat.dat   67584 11/11/2025 1:38:37 PM
twain_32.dll   69120 4/1/2024 11:43:03 AM
bfsvc.exe      102400 6/7/2025 6:33:43 PM
splwow64.exe   224256 6/7/2025 6:34:11 PM
WHSysPr9.prx   316640 4/1/2024 11:43:04 AM
notepad.exe    327680 6/7/2025 6:34:08 PM
regrid.exe     587264 6/7/2025 6:34:13 PM
HelpPane.exe   1078784 6/7/2025 6:33:57 PM
explorer.exe   2849072 6/7/2025 6:33:55 PM

PS C:\Windows> Get-ChildItem -File |
>> Where-Object { $_.Length -gt 10000 } |
>> Sort-Object Length |
>> Select-Object Name, Length, LastWriteTime |
>> Out-File C:\Lab3_WindowsFiles.txt
```

## 5. Получить процессы с приоритетом больше 7

```
PS C:\Windows> Get-Process |
>> Where-Object { $_.BasePriority -gt 7 } |
>> Sort-Object ProcessName |
>> Select-Object ProcessName, BasePriority, Company |
>> Format-Table
ProcessName      BasePriority Company
-----          -----  -----
AggregatorHost      8
ApplicationFrameHost 8 Microsoft Corporation
coherence          8
coherence          8 Microsoft Corporation
conhost            8
cssrss             13
cssrss             13
ctfmon             13
ctfmon             13
dllhost            8 Microsoft Corporation
dllhost            8 Microsoft Corporation
dmw               13
explorer           8 Microsoft Corporation
fontdrvhost        8
fontdrvhost        8
lsass              9
Memory Compression 8
MicrosoftEdgeUpdate 8
MohotificationNx  8 Microsoft Corporation
MpDefenderCoreService 8
msedgewebview2     8 Microsoft Corporation
```

## 6-7. Создание HTML-файла с процессами

```
PS C:\Windows> Get-Process |
>> Where-Object { $_.BasePriority -gt 7 } |
>> Sort-Object ProcessName |
>> Select-Object ProcessName, BasePriority, Company |
>> ConvertTo-HTML | Out-File "Список процессов (BasePriority > 7).html" |
>> Out-File C:\Lab3_Processes.html
PS C:\Windows>
```

## 8. Суммарный объём всех графических файлов (bmp, jpg)

```
PS C:\Windows> Get-ChildItem -Path C:\Windows -Include *.bmp, *.jpg -Recurse -ErrorAction SilentlyContinue |
>> Measure-Object -Property Length -Sum
Count : 195
Average : 92503699
Sum : 92503699
Maximum : 3072
Minimum : 0
Property : Length
```

## 9. Сведения о ЦП

```
PS C:\Windows> Get-CimInstance Win32_Processor | Select-Object Name, NumberOfCores, MaxClockSpeed
>>
Name      NumberOfCores MaxClockSpeed
Apple Silicon       4          3072
```

## 10. Время выполнения команд

```
PS C:\Windows> Measure-Command { dir } | Select-Object TotalMilliseconds
>> Measure-Command { Get-Process } | Select-Object TotalMilliseconds
>>
TotalMilliseconds
-----
16.9067
6.0737
```

```
PS C:\Windows> $N = 10
>>
>> $dirTimes = @()
>> $procTimes = @()
>>
>> for ($i = 1; $i -le $N; $i++) {
>>   $t = (Measure-Command { dir > $null }).TotalMilliseconds
>>   $dirTimes += $t
>> }
>>
>> for ($i = 1; $i -le $N; $i++) {
>>   $t = (Measure-Command { Get-Process > $null }).TotalMilliseconds
>>   $procTimes += $t
>> }
>>
>> Write-Host " Время выполнения команды DIR "
>> Write-Host ("Максимум: {0:N2} мс" -f ($dirTimes | Measure-Object -Maximum).Maximum)
>> Write-Host ("Минимум: {0:N2} мс" -f ($dirTimes | Measure-Object -Minimum).Minimum)
>> Write-Host ("Среднее: {0:N2} мс" -f ($dirTimes | Measure-Object -Average).Average)
>>
>> Write-Host " Время выполнения команды Get-Process "
>> Write-Host ("Максимум: {0:N2} мс" -f ($procTimes | Measure-Object -Maximum).Maximum)
>> Write-Host ("Минимум: {0:N2} мс" -f ($procTimes | Measure-Object -Minimum).Minimum)
>> Write-Host ("Среднее: {0:N2} мс" -f ($procTimes | Measure-Object -Average).Average)
>>
>> Время выполнения команды DIR
Максимум: 17.89 мс
Минимум: 2.01 мс
Среднее: 3.94 мс
>>
>> Время выполнения команды Get-Process
Максимум: 2.61 мс
Минимум: 0.59 мс
Среднее: 0.90 мс
PS C:\Windows>
```

## 11. Вычисление факториала числа.

Нахождение минимального и максимального значения из файла nn.txt.

```
PS C:\Windows> $num = Read-Host "Введите число для вычисления факториала"
>> $factorial = 1
>>
>> for ($i = 1; $i -le $num; $i++) {
>>   $factorial *= $i
>> }
>>
>> Write-Host "Факториал числа $num = $factorial"
Введите число для вычисления факториала: 6
Факториал числа 6 = 720
```

```
PS C:\Windows> "12 n45 n7 n89 n23" | Out-File C:\nn.txt
>>
PS C:\Windows> Get-Content C:\nn.txt
>>
12
45
7
89
23
```

```
PS C:\Windows> $numbers = Get-Content C:\nn.txt | ForEach-Object { [int]$_. }
>>
>> $min = ($numbers | Measure-Object -Minimum).Minimum
>> $max = ($numbers | Measure-Object -Maximum).Maximum
>>
>> Write-Host "Минимальное значение: $min"
>> Write-Host "Максимальное значение: $max"
Минимальное значение: 7
Максимальное значение: 89
```