

Практическая работа 17

Диспетчер задач

Диспетчер задач Windows — диспетчер задач, системный монитор и менеджер запуска, входящий в состав Windows. Он предоставляет информацию о производительности компьютера и запущенных приложениях, процессах и использовании ЦП, фиксирует нагрузку и сведения о памяти, сетевой активности и статистике, зарегистрированных пользователях и системных службах. Диспетчер задач также может использоваться для установки приоритетов процессов, свойства процессора, запуска и остановки служб и принудительного завершения процессов.


Используя методы класса `Process`(<https://learn.microsoft.com/ru-ru/dotnet/api/system.diagnostics.process?view=net-7.0>) разработать собственный диспетчер задач. В программе необходимо отобразить общее количество запущенных процессов, их список (название и количество используемой памяти). Предусмотреть механизм завершения процесса. Обновление состояния системы реализовать 1 раз в секунду

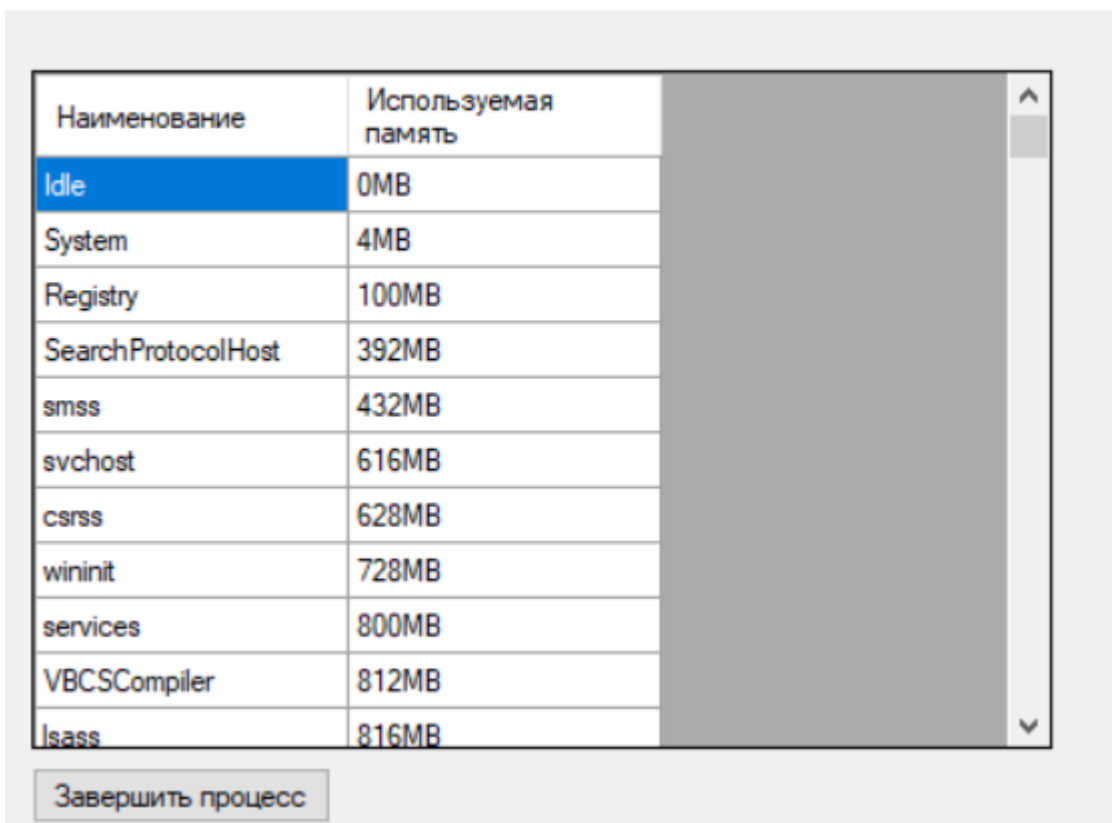
Form1

Наименование	Используемая память
Idle	0MB
System	4MB
Registry	100MB
SearchFilterHost	240MB
SearchProtocolHost	392MB
smss	432MB
svchost	616MB
csrss	628MB
wininit	728MB
services	800MB
VBCSCompiler	812MB

Завершить процесс

Рисунок 1 – Интерфейс программы

 Form1



Наименование	Используемая память
Idle	0MB
System	4MB
Registry	100MB
SearchProtocolHost	392MB
smss	432MB
svchost	616MB
csrss	628MB
wininit	728MB
services	800MB
VBCSCompiler	812MB
lsass	816MB

Завершить процесс

Рисунок 2 – Обновление

 Form1

Наименование	Используемая память
WINWORD	10660MB
ServiceHub.Host.CLR....	10724MB
conhost	10864MB
fontdrvhost	10964MB
NVDisplay.Container	10984MB
chrome	11016MB
explorer	11044MB
svchost	11252MB
svchost	11260MB




Рисунок 3 – Завершение процесса chrome

Наименование	Используемая память
svchost	10492MB
WINWORD	10660MB
ServiceHub.Host.CLR....	10724MB
conhost	10864MB
fontdrvhost	10964MB
NVDisplay.Container	10984MB
explorer	11044MB
svchost	11252MB
svchost	11260MB

Завершить процесс

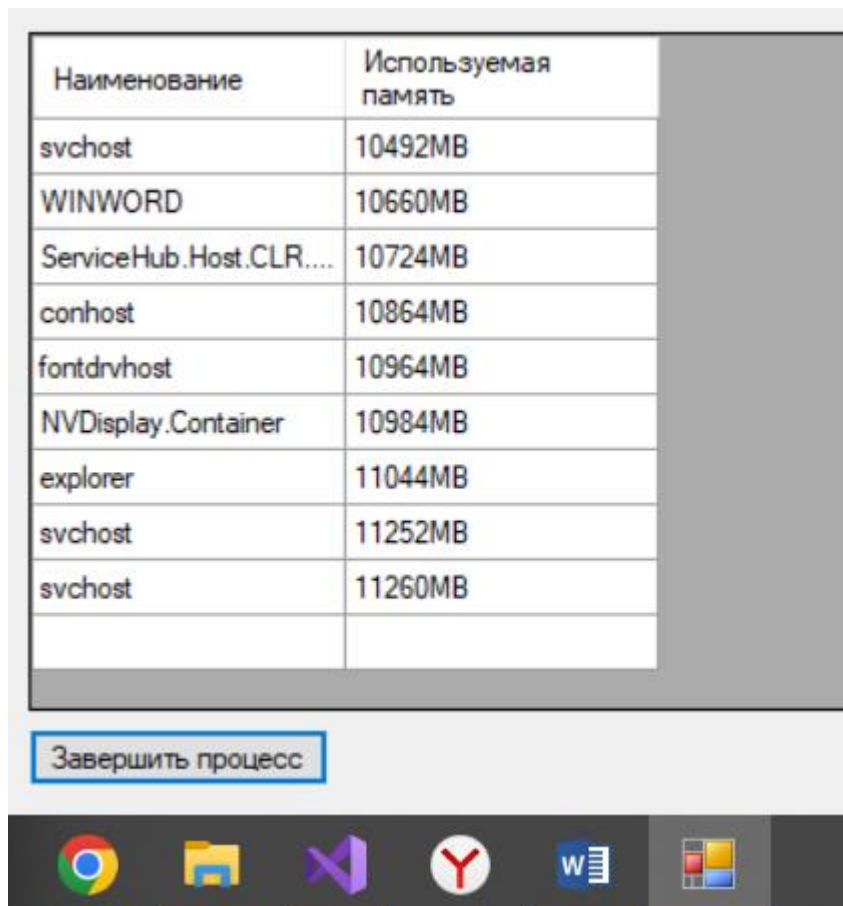


Рисунок 4 – Результат завершения процесса

Листинг

```

public void dispatcher()
{
    timer1.Interval = 10000;
    timer1.Start();
}
private void Form1_Load(object sender, EventArgs e)
{
    dispatcher();
}

private void timer1_Tick(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    var allProcess = from pr in Process.GetProcesses(".")
                    orderby pr.Id
                    select pr;
    foreach (var proc in allProcess)
    {
        string[] arr = { "" + proc.ProcessName, "" + proc.Id + "MB",
                        "" + proc.WorkingSet64 / 1000000 + " MB", "" +
proc.VirtualMemorySize64 / 1000000 + "MB", "" + proc.MachineName, "" +
proc.BasePriority};
        dataGridView1.Rows.Add(arr);
    }
}

private void button1_Click(object sender, EventArgs e)
{

```

```
var z = dataGridView1.CurrentRow.Cells[0].Value.ToString();  
foreach (var process in Process.GetProcessesByName(z))  
{  
    process.Kill();  
}  
}
```

Ссылка на гитхаб:

<https://github.com/Alexandrov911/PR17Sistem.git>