

Управление процессами

Пример

При запуске системы запускается множество процессов, о которых пользователь зачастую даже и не подозревает. В дополнение к этому могут быть запущены процессы, инициированные пользователем в явном виде. В течение 1 секунды он может успеть поработать с несколькими из них, создавая иллюзию параллельной работы. Иногда в этом случае говорят о псевдопараллелизме

Что такое процесс

Процесс — это просто экземпляр выполняемой программы, включая текущие значения счетчика команд, регистров и переменных.

Что такое процесс

Процесс — это своего рода действия. У него есть программа, входные, выходные данные и состояние. Один процессор может совместно использоваться несколькими процессами в соответствии с неким алгоритмом планирования, который используется для определения, когда остановить один процесс и обслужить другой

Процесс и программа

В отличие от процесса программа может быть сохранена на диске и вообще ничего не делать.

Программа запущена несколько раз

Если программа запущена дважды, то считается, что ею заняты два процесса. Например, зачастую возможно дважды запустить текстовый процессор или одновременно распечатать два файла, если одновременно доступны два принтера. Тот факт, что два работающих процесса запущены от одной и той же программы, во внимание не принимается, поскольку это два разных процесса.

Создание процесса

События, приводящие к созданию процессов:

- Инициализация системы.
- Выполнение работающим процессом системного вызова, предназначенного для создания процесса.
- Запрос пользователя на создание нового процесса.
- Инициация пакетного задания.

Завершение процесса

События, приводящие к завершению процессов:

- обычного выхода (добровольно);
- выхода при возникновении ошибки (добровольно) вызова, предназначенного для создания процесса.
- возникновения фатальной ошибки (принудительно);
- уничтожения другим процессом (принудительно).

Иерархии процессов

В некоторых ОС, когда процесс порождает другой процесс, родительский и дочерний процессы продолжают быть определенным образом связанными друг с другом. Дочерний процесс может и сам создать какие-нибудь процессы, формируя иерархию процессов

Состояния процессов

- 1) выполняемый (использующий в данный момент центральный процессор);
- 2) готовый (работоспособный, но временно приостановленный, чтобы дать возможность выполнения другому процессу);
- 3) заблокированный (неспособный выполняться, пока не возникнет какое-нибудь внешнее событие)

Состояния процессов



1. Процесс заблокирован в ожидании ввода
2. Диспетчер выбрал другой процесс
3. Диспетчер выбрал данный процесс
4. Входные данные стали доступны

Работа с процессами

В .NET процесс представлен классом `Process` из пространства имен `System.Diagnostics`. Этот класс позволяет управлять уже запущенными процессами, а также запускать новые.

Работа с процессами

Свойства класса Process:

- **Handle:** возвращает дескриптор процесса
- **Свойство Id:** получает ID процесса
- **Свойство MachineName:** возвращает имя компьютера, на котором запущен процесс
- **Свойство ProcessName:** возвращает имя процесса
- **StartTime:** время запуска процесса

Работа с процессами

Методы класса Process:

- `CloseMainWindow()`: закрывает окно процесса
- `GetProcesses()`: возвращает массив всех запущенных процессов
- `GetProcessesByName()`: возвращает процессы по его имени.

Работа с процессами

Методы класса Process:

- `GetProcessById()`: возвращает процесс по Id.
- `Kill()`: останавливает процесс
- `Start()`: запускает новый процесс

Пример

```
var proc = Process.GetCurrentProcess();  
Console.WriteLine($"Id: {proc.Id}");  
Console.WriteLine($"Name: {proc.ProcessName}");  
Console.WriteLine($"Memory: {proc. PageMemorySize64}");
```


Пример

```
foreach (Process process in Process.GetProcesses()){  
    // выводим id и имя процесса  
    Console.WriteLine($"ID: {process.Id} Name: {process.ProcessName}");  
}
```

Пример

```
Process.Start(@"C:\Program Files\Google\Chrome\Application\chrome.exe");
```

```
Process.Start("notepad.exe");
```

Пример

```
ProcessStartInfo procInfo = new ProcessStartInfo();
```

```
// исполняемый файл программы - браузер хром
```

```
procInfo.FileName = @"C:\Program Files\Google\Chrome\  
Application\chrome.exe";
```

```
// аргументы запуска - адрес интернет-ресурса
```

```
procInfo.Arguments = "https://arcotel.ru/";
```

```
Process.Start(procInfo);
```

Выполнение консольной команды

```
string cmd = "calc.exe";  
var proc = new ProcessStartInfo(){  
    UseShellExecute = true,  
    FileName = @"C:\Windows\System32\cmd.exe",  
    Arguments = "/c " + cmd,  
    WindowStyle = ProcessWindowStyle.Hidden  
};  
Process.Start(proc);
```

Задание

- Вывести все запущенные процессы
- Вывести процесс с наибольшим номером PID
- Запустить калькулятор (calc.exe)
- Найти процесс, соответствующий запущенному калькулятору и вывести информацию о нём

Изменение приоритета у всей программы

// Получаем текущий процесс

```
Process currentProcess = Process.GetCurrentProcess();
```

// Устанавливаем наивысший приоритет для процесса

```
currentProcess.PriorityClass = ProcessPriorityClass.Realtime;
```

```
Console.WriteLine("Приоритет процесса установлен на Realtime");
```

// Выполнение остальной части программы..