

Модель эпидемии

Дано

- Есть вирус, вероятность заражения которым при контакте – 0,2
- Инкубационный период – 3 дня
- После этого пациент болеет 7 дней
- Через 7 дней пациент умирает с вероятностью 0,1
- Переболевшие приобретают иммунитет
- За время болезни пациент распространяет вирус

Класс вирус

- Вероятность заражения
- Инкубационный период
- Болезнь
- Вероятность летального исхода

Класс пациент

- Идентификатор
- Состояние
 - Здоров
 - Носитель вируса в инкубационной стадии
 - Болен
 - Приобрел иммунитет
 - Умер

Класс пациент

- Идентификатор
- Жив/мертв
- Заражен / не заражен
- Срок от начала заражения
 - По вирусу определяем:
 - Инкубационный период
 - Открытое течение болезни
- Если пациент жив после болезни, то он обладает иммунитетом

Класс «Много пациентов»

- Отражает состояние пациентов на определенном шаге во времени

Класс «Эпидемия»

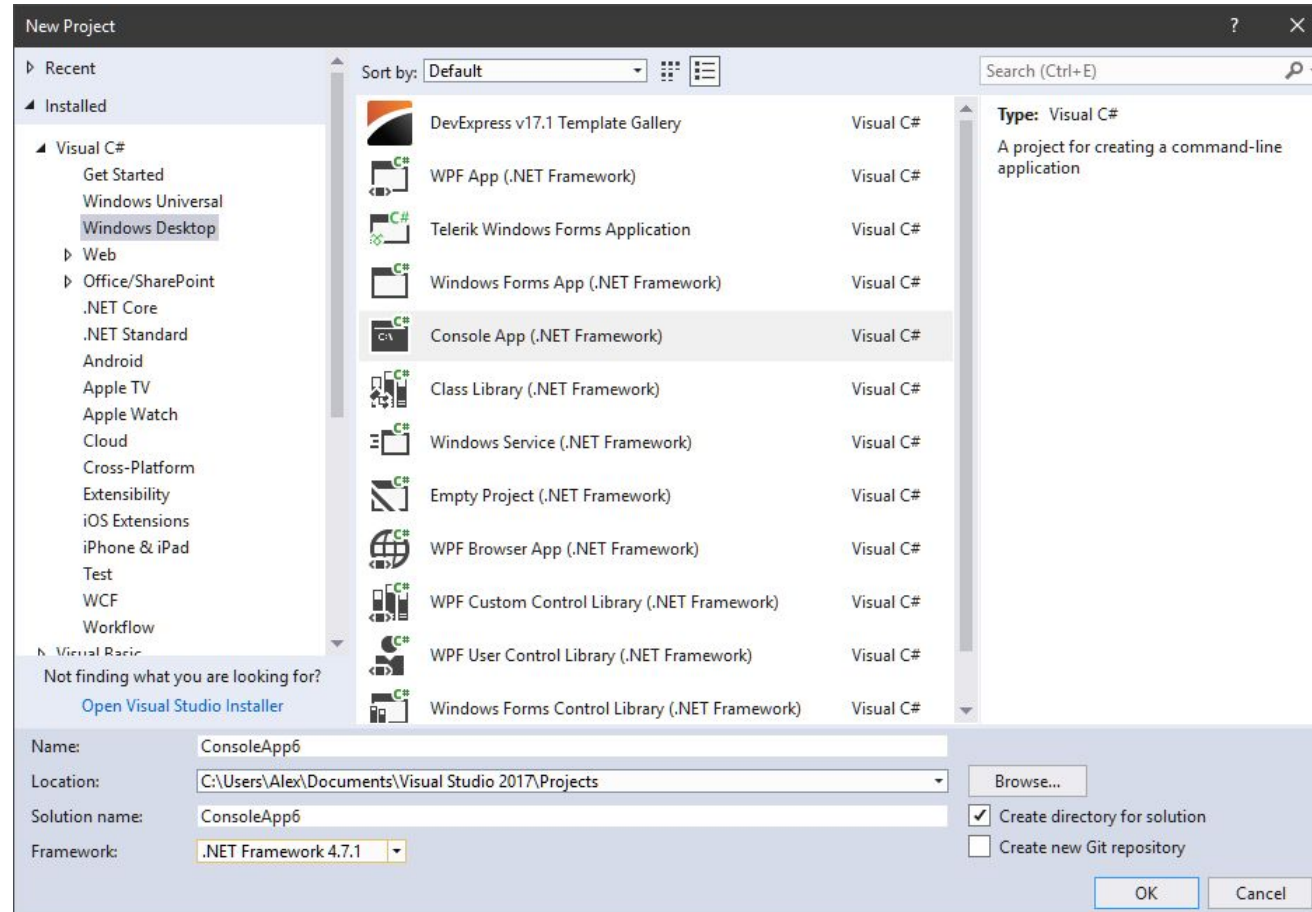
- Считает время
- На каждом шаге считает:
 - Новых зараженных
 - Меняет сроки болезни зараженным
 - Убивает / выздоравливает
- Считает статистику

Общий алгоритм

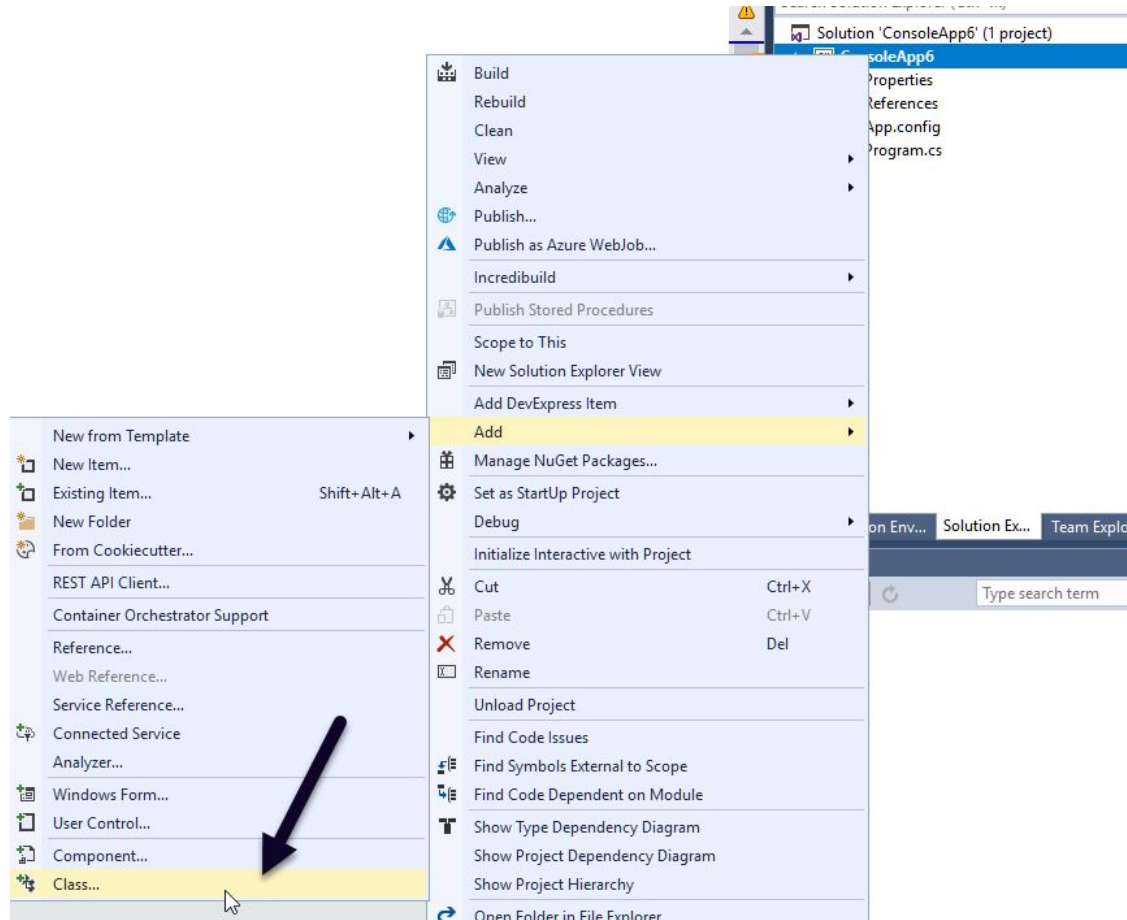
- Создаем вирус
- Создаем пациентов
- «Заражаем» нескольких
- Запускаем цикл:
 - Перемешиваем пациентов
 - «Заражаем» контактирующих, если у них нет иммунитета и они живы
 - У зараженных – увеличиваем срок болезни на 1
 - Убиваем переболевших или придаем им иммунитет
 - Считаем статистику
 - Повторяем

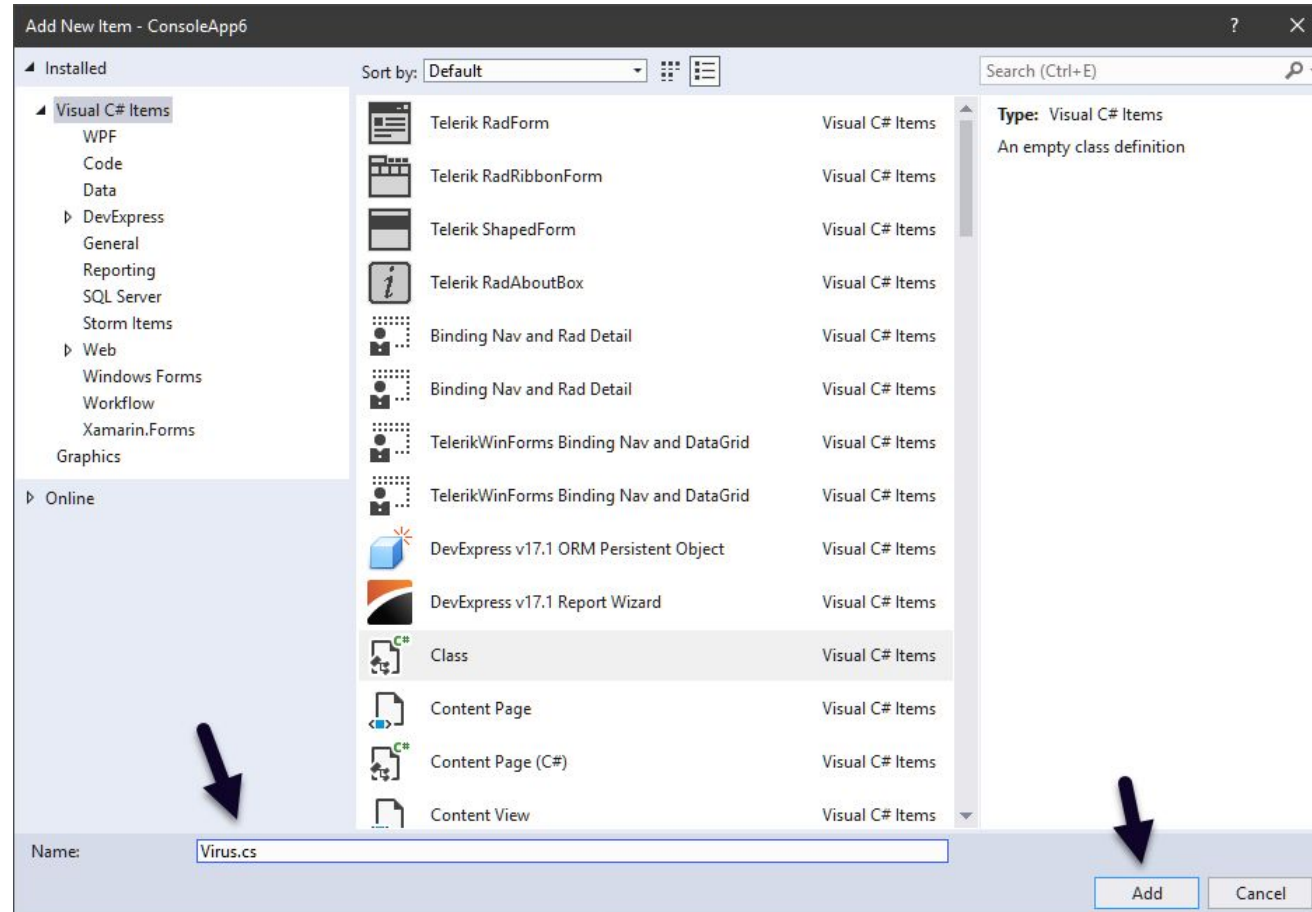
Как перемешать пациентов и найти контактирующих?

- Один из вариантов! (Возможны и другие)
- Пробегаем по пациентам и находим зараженного
- Случайным образом генерируем 5-10 номеров контактирующих
- Если они живы и без иммунитета – заражаем их с заданной вероятностью



- namespace ConsoleApp6
- {
- class Program
- {
- static void Main(string[] args)
- {
- Console.WriteLine("Эпидемия");
- Console.ReadKey();
- }
- }
- }

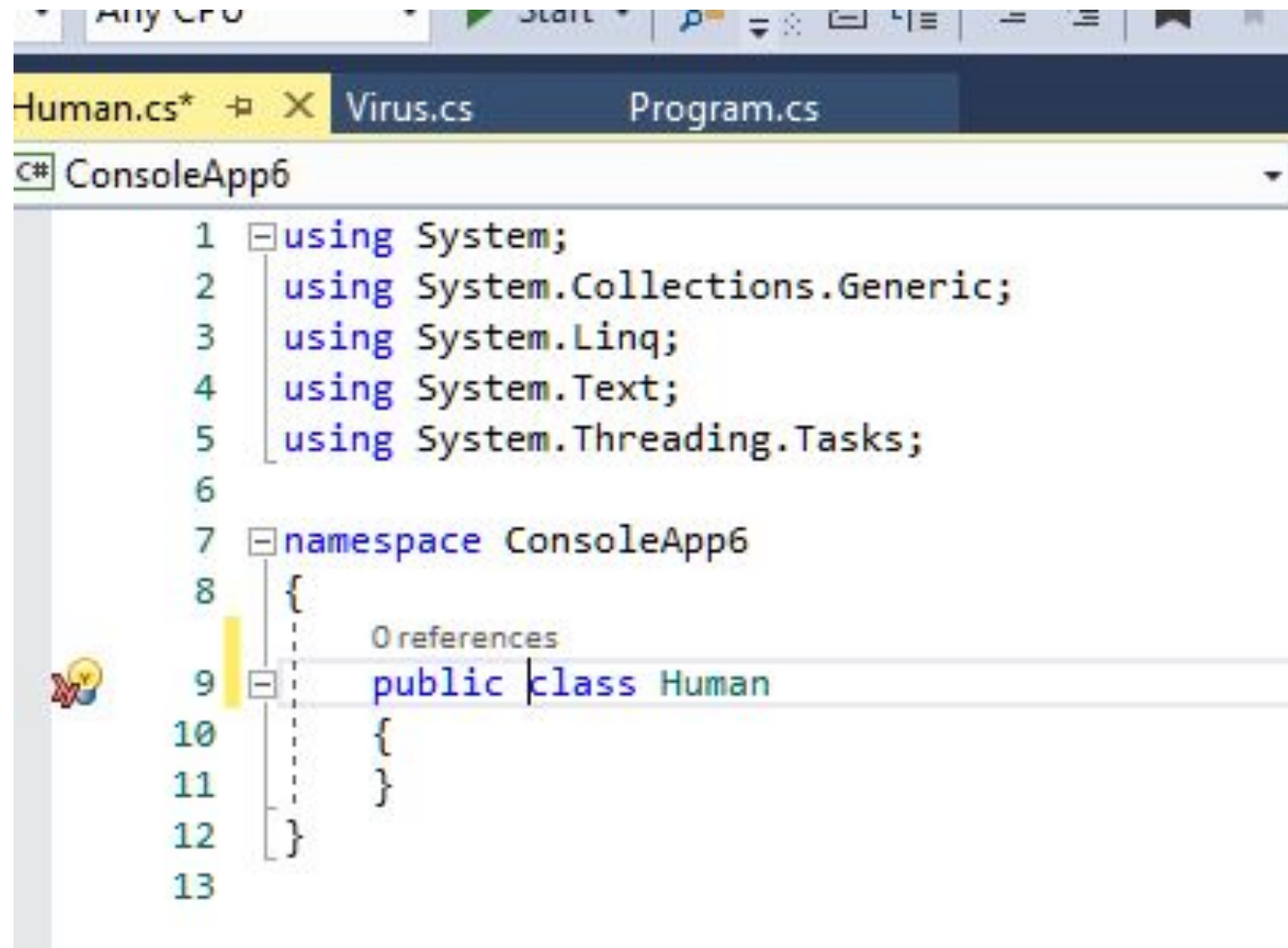




Понятие о классе и свойствах

```
9 public class Virus
10 {
11     // вероятность заражения
12     1 reference
13     public double Zaraza { get; set; }
14     // инкубационный период
15     1 reference
16     public int IncTime { get; set; }
17     // время болезни
18     1 reference
19     public int BolesnTime { get; set; }
20     // вероятность смерти
21     1 reference
22     public double Dead { get; set; }
23
24     0 references
25     public Virus(double zaraza = 0.2, int incTime = 3, int bolesnTime = 7, double dead = 0.1)
26     {
27         Zaraza = zaraza;
28         IncTime = incTime;
29         BolesnTime = bolesnTime;
30         Dead = dead;
31     }
32 }
```

Класс пациент



The image shows a screenshot of the Visual Studio IDE. The top toolbar includes buttons for 'Any CPU', 'Start', and other standard development tools. The window title bar shows three open files: 'Human.cs*', 'Virus.cs', and 'Program.cs'. The active file is 'Human.cs', which is part of a 'C# ConsoleApp6' project. The code in the editor is as follows:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace ConsoleApp6
8 {
9     public class Human
10    {
11    }
12 }
13
```

A yellow vertical bar highlights line 9, and a tooltip is visible over the 'public class Human' declaration, showing '0 references'.

0 references

```
public class Human
```

```
{
```

0 references

```
public int id { get; set; } = 0;
```

```
// мертвый или живой
```

0 references

```
public bool IsDead { get; set; } = false;
```

```
// если живой, то здоровый или нет?
```

0 references

```
public bool IsAlive { get; set; } = true;
```

```
// время заражения
```

0 references

```
public int InfestTime { get; set; } = 0;
```

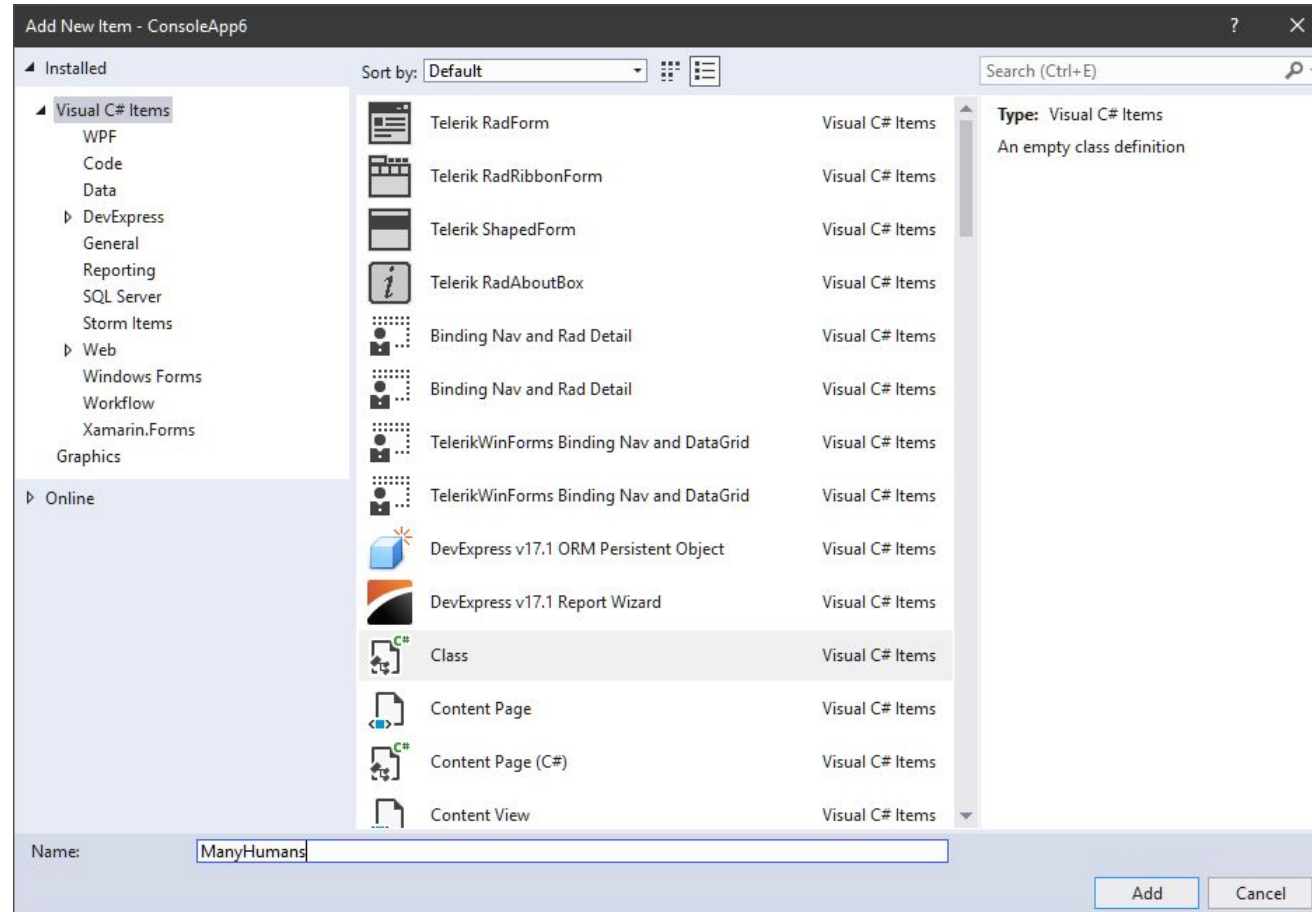
```
// есть ли иммунитет
```

0 references

```
public bool HaveImmunitet { get; set; } = false;
```

```
}
```


Много пациентов



Добавлен метод для заполнения массива пациентов

```
0 references
public class ManyHumans
{
    // всего жителей
    1 reference
    public int MAX_HUMANS { get; set; } = 200000;
    1 reference
    public List<Human> Humans { get; set; } = new List<Human>();

    0 references
    public void Init()
    {
        for (int i = 0; i < this.MAX_HUMANS; i++)
        {
            Human h = new Human();
            h.id = i;
            Humans.Add(h);
            Console.WriteLine($"Пациент {i}");
        }
    }
}
```

```
6
7 namespace ConsoleApp6
8 {
9     class Program
10    {
11        static void Main(string[] args)
12        {
13            Console.WriteLine("Эпидемия");
14
15            ManyHumans mh = new ManyHumans();
16            mh.Init();
17
18            Console.ReadKey();
19        }
20    }
21 }
```