В файле «Vipusknaya2» помимо размера поля и начального количества М можно выбрать условие, при которых М остается в живых:

1. Количество соседей:

Когда при каждой итерации изменяется промежуток количества соседей, когда М остается в живых, на пример:

> 1 и < 4 – означает, что М жив, когда кол-во соседей = 2 или 3

2) Расположение соседей:

Когда при каждой итерации изменяется расположения соседей, когда М остается в живых;

Есть такие виды расположения (С – сосед, М – изучяемый М):

1.

С

М

С

2.

С

М

С

3.

С

С

М

4.

С

С

С

М

Условия рождения М остается одинаковым при обоих вариантах условий выживаний:

С каждой итерацией изменяется переменная q1;

Когда у пустой клетки соседей = q1 – рождается М;

int q1 = r.Next(2, 5); - объявление ее перед каждой итерацией

В циклах:

if (a[i, j] == 1)//если здесь М

{

if (location\_of\_neigh(f, i, j) == true)

//если условия виживания положительные, то М остается

{

g.FillRectangle(Brushes.Gold, i \* z, j \* z, z, z);

g.DrawRectangle(Pens.Red, i \* z, j \* z, z, z);

}

else//в инном случае М умирает

{

d++;

a[i, j] = 0;

g.FillRectangle(Brushes.White, i \* z, j \* z, z + 1, z + 1);

}

}

//если это пустая клетка и условие рождения совпадает:

else if (num\_of\_neigh(i, j) == q1)

{

d--;

a[i, j] = 1;

g.FillRectangle(Brushes.Gold, i \* z, j \* z, z, z);

g.DrawRectangle(Pens.Red, i \* z, j \* z, z, z);

}

В программе есть кнопка «График», которая вырисовывает зависимость количества М от времени:

for (int i = 0; i < abc + 1; i++)//находим максимальное значение для масштаба:

{

if (max < n[i])

max = n[i];

}

x1 = 0;

y1 = pictureBox2.Height;

for (int i = 0; i < abc + 1; i++)

{

g1.DrawLine(Pens.Red, x1, y1, i \* pictureBox2.Width / abc, pictureBox2.Height - 5 - (n[i]) \* pictureBox2.Height / max);

x1 = i \* pictureBox2.Width / abc;

y1 = pictureBox2.Height - 5 - (n[i]) \* pictureBox2.Height / max;

g1.DrawEllipse(Pens.Black, x1 - 5, y1 - 5, 10, 10);//

}