

```
D:\dilema\Debug\dilema.exe
Game 13
11001011111111111111111111111111 Fitness= 65 - 1
01001000010101010101010101010101 Fitness= 33 - 0
Game 14
11110000010111010101010101010101 Fitness= 44 - 0
10000110010011110110101010101010 Fitness= 48 - 1
Game 15
00111100001011000010101010101010 Fitness= 43 - 0
01010010011100100111100100111001 Fitness= 51 - 1
Game 16
01111100000111100110101010101010 Fitness= 49 - 1
00010110010011011011000100110010 Fitness= 45 - 0
Game 17
10111010111010111010101010101010 Fitness= 54 - 1
10110100110100110100110100110100 Fitness= 42 - 0
Game 18
01100000110011010011101010101010 Fitness= 47 - 1
11011000000010010010101010101010 Fitness= 39 - 0
Game 19
00101111111111111111111111111111 Fitness= 50 - 1
01101111111111111111111111111111 Fitness= 50 - 1
Game 20
00000000000000000000000000000000 Fitness= 34 - 1
00000000000000000000000000000000 Fitness= 34 - 1
Для продолжения нажмите любую клавишу . . .
```

```
#include "stdafx.h"
```

```
#include <iostream>
```

```
#include <stdlib.h>
```

```
#include <ctime>
```

```
using namespace std;
```

```
void main()
```

```
{
```

```
    int bit[40][64];
```

```
    int a[22], b[22], asum=0, bsum=0;
```

```
    int players[40][22];
```

```
    int i,j;
```

```
    //srand(11);
```

```
    for(j=0; j<40; j++)
```

```
    {
```

```
        for (i=0; i<64; i++)
```

```
        {
```

```
            bit[j][i]=rand()%2;
```

```
        }
```

```

    }

    int game=1;
    for (int k=0; k<40; k++)
    {
        cout<<"Game " <<game<<endl;
        game++;

        for (i=0; i<3; i++)
        {

            players[k][i]=rand()%2;
            players[k+1][i]=rand()%2;
        }

        /*for (i=0; i<3; i++)
        {
            cout<<players[k][i];

        }
        cout<<endl;

        for (i=0; i<3; i++)
        {
            cout<<players[k+1][i];

        }
        cout<<endl<<endl;*/
        players[k][20]=0;
        players[k+1][20]=0;
        for(i=3; i<20; i++)
        {
            int abit=0,bbit=0;
            abit=players[k+1][i-1]*32+players[k][i-
1]*16+players[k+1][i-2]*8+players[k][i-2]*4+players[k+1][i-3]*2+players[k][i-3];
            bbit=players[k][i-1]*32+players[k+1][i-
1]*16+players[k][i-2]*8+players[k+1][i-2]*4+players[k][i-3]*2+players[k+1][i-3];
            players[k][i]=bit[k][abit];
            players[k+1][i]=bit[k+1][bbit];
            if (players[k][i]==1 && players[k+1][i]==1)
            {
                players[k][20]+=3;

```

```

        players[k+1][20]+=3;
    }
    if (players[k][i]==1 && players[k+1][i]==0)
    {
        players[k][20]+=5;
        players[k+1][20]+=1;
    }
    if (players[k][i]==0 && players[k+1][i]==1)
    {
        players[k][20]+=1;
        players[k+1][20]+=5;
    }

    if (players[k][i]==0 && players[k+1][i]==0)
    {
        players[k][20]+=2;
        players[k+1][20]+=2;
    }
}

if (players[k][20]>players[k+1][20])
{
    players[k][21]=1;
    players[k+1][21]=0;
}
if (players[k][20]<players[k+1][20])
{
    players[k][21]=0;
    players[k+1][21]=1;
}
if (players[k][20]==players[k+1][20])
{
    players[k][21]=1;
    players[k+1][21]=1;
}

for (i=0; i<20; i++)
{
    cout<<players[k][i];
}

```

```

        cout<<"  Fitness= "<<players[k][20]<<" -
"<<players[k][21]<<endl;

        for (i=0; i<20; i++)
        {
            cout<<players[k+1][i];

        }
        cout<<"  Fitness= "<<players[k+1][20]<<" -
"<<players[k+1][21]<<endl;
        k++;
    }
    system("pause");
}

```