**Lab4 Simulation :**

**SIMULATION OF RABBIT POPULATION GROWTH**

Mouncet Mohammed Amine

El Allali Achraf

1. Rabbit population simulation : Fibonacci method :

For the first experience of the simulation, we used Fibonacci numbers to generate rabbits family tree

1. Code:

int fibonnaci(int N) {

    if (N == 0 || N == 1) {

        return 2;

    }

    else {

        return fibonnaci(N - 1) + fibonnaci(N - 2);

    }

}

We used recursive technic to create a Fibonacci function in C language, it consists of a sequence of Fibonacci’s function calls inside itself, with the adjustment of the parameter N, which represent the number of steps in Fibonacci numbers. The Fibonacci numbers used in this function starts with 2, 2 comparing to the original one tha starts with (0, 1), because in our experience, we need at least a couple to simulate a population, wich modify the original Fibonacci numbers but apply the same method of calculation.

1. Results:

We supposed that the rabbits tree starts with an immature couple (N=0) and by the first step (N=1) they became mature, which makes us assume that the first step is between 5 and 8 months in real life. By the second step (N=2), and following Fibonacci’s calculation, the first couple produce their first pair, making a total population of 4.

The following figure shows a sequence of 23 generation (steps) using Fibonacci function.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| STEP | POPULATION | STEP | POPULATION | STEP | POPULATION | STEP | POPULATION |
| 0 | 2 | 6 | 26 | 12 | 466 | 18 | 8362 |
| 1 | 2 | 7 | 42 | 13 | 754 | 19 | 13530 |
| 2 | 4 | 8 | 68 | 14 | 1220 | 20 | 21892 |
| 3 | 6 | 9 | 110 | 15 | 1974 | 21 | 35422 |
| 4 | 10 | 10 | 178 | 16 | 3194 | 22 | 57314 |
| 5 | 16 | 11 | 288 | 17 | 5168 | 23 | 92736 |

1. Analysis:

The main problem with the Fibonacci rabbits’ tree is that we don’t take into consideration the death of a rabbit, as shown in the function, the rabbits generated are immortal, which is far from reality as the life expectancy of a rabbit is about 9 years, which lead us to a population of zombie rabbits that are dead but still giving birth.

The second problem is the unrealistic steps and generation duration, as explained before, each step for an immature pair (the first generation and every new born) represent their hole maturity phase which is 5 to 8 months, at the same time, one step represent to a mature pair their pregnancy and giving birth of pair, the duration in real life is one month, so one step is time irrational.

There are many other probability problems as the fact of obtaining in each litter two baby rabbits, one male and one female which is far from the real statistics.

1. Rabbit population simulation: More realistic population growth:

For the second part, we will be using a more realistic program to generate the rabbit population, using life observed statistic and probability, to try and solve some of the problems observed in the last experience.

1. Code :

typedef struct Rabbit {

    char sex;            // M: Male, F: Female

    int status;           // 0:dead ou 1:alive

    int age;              // 0:just born, -1 dead? (age in months)

    int mature;           // x0:no, x1:yes(adult) x1: at age x become mature

    int pregnant;         // 0:no, 1:yes

    int nbLittersY;       // litters must have in a year

    int nbLitters;        // litters made so far

    int srvRate; // survive rate <= 100%, <0 means has been used, >=0 will be used (is positif when created ot updated)

    struct Rabbit\* nextRabbit;  // next rabbit

}Srabbit;

During this we will consider the rabbit tree as a linked list sorted by generation, more precisely every brother and sister will be linked closely to each other, and at the same time followed by the other siblings on the same generation.

Every element of the list, that represent a rabbit, is a C structure that contain 9 parameters: rabbit information’s parameters like sex, status, age, maturity, pregnant, and probability parameters nbLittersY that represent the yearly number of litters that will be generated, nbLitters that shows how many litter the rabbit did for that year, the srvRate for survive rate and finaly a pointer to the next rabbit in the list.

The probabilities in the code works monthly but with a yearly cycle, as for the number of litter that each female will do this year is calculated for the first time in the month she became mature, we will calculate how many litter a female will do that year and then we will distribute that number of litters on the 12 next months, at the end of the year a new number will be generated for her second maturity year. As for survival rate, it will be recalculated and applied once every year.

1. Results:

The program take two parameters, the first for the number of rabbits of the first generation, and the second one the duration in months of the experience