**Programming Question**

The assignment is to code a function/method in the form of “RabbitCount(int month)” that returns the number of Rabbit Pairs we would have in a given month based on the following rules about Rabbit lifecycles:

* At any month before 0 there are no Rabbits in existence
* At month 0 there magically is 1 pair of Rabbits
* Rabbits always come in pairs
* Other than the first pair of Rabbits, no other Rabbits may magically appear
* A pair of Rabbits take 3 months from the time they are born to mature
* On the month a pair of Rabbits mature they give birth to their first pair of babies
* Mature Rabbit Pairs give birth to a pair of babies every month
* Rabbits never die

Sample test input:

|  |  |
| --- | --- |
| **Month** | **Expected Output** |
| 0 | 1 |
| 3 | 2 |
| 6 | 6 |
| 9 | 19 |
| 12 | 60 |

The expected result of this test is a function that can be called to determine the number of Rabbit Pairs we would have given in input of the number of Months that have passed.

// Please find the code in C#.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Fleetmatics

{

class Rabbits

{

internal static List<RabbitPair> rabbits = new List<RabbitPair>();

public static int RabbitCount(int month)

{

//add magically appeared pair of rabbits

if (month >= 0)

{

rabbits.Add(new RabbitPair());

}

for (int m = 1; m <= month; m++)

{

IEnumerator<RabbitPair> itr = rabbits.GetEnumerator();

int newPairs = 0;

while (itr.MoveNext())

{

RabbitPair rp = itr.Current;

rp.increaseAgeByAMonth();

if (rp.Mature)

{

//Rabbit pair is mature to give birth to babies

newPairs++;

}

}

for (int i = 0; i < newPairs; i++)

{

rabbits.Add(new RabbitPair());

}

}

return rabbits.Count;

}

public static void Main(string[] args)

{

Console.WriteLine(RabbitCount(6));

Console.Read();

}

}

internal class RabbitPair

{

internal int age = 0;

public virtual void increaseAgeByAMonth()

{

age++;

}

public virtual int Age

{

set

{

this.age = value;

}

get

{

return age;

}

}

public virtual bool Mature

{

get

{

return (age >= 3) ? true : false;

}

}

}

}

// Please find the code in JAVA.

**import** java.util.ArrayList;

**import** java.util.Iterator;

**public** **class** RabbitCount {

**static** ArrayList<RabbitPair> *rabbits* = **new** ArrayList<RabbitPair>();

**public** **static** **int** RabbitCount(**int** month){

//add magically appeared pair of rabbits

**if**(month >= 0)

*rabbits*.add(**new** RabbitPair());

**for**(**int** m = 1; m <= month; m++){

Iterator<RabbitPair> itr = *rabbits*.iterator();

**int** newPairs = 0;

**while**(itr.hasNext()){

RabbitPair rp = itr.next();

rp.increaseAgeByAMonth();

**if**(rp.isMature()){

//Rabbit pair is mature to give birth to babies

newPairs++;

}

}

**for**(**int** i = 0; i < newPairs; i++){

*rabbits*.add(**new** RabbitPair());

}

}

**return** *rabbits*.size();

}

**public** **static** **void** main(String[] args){

System.*out*.println(*RabbitCount*(7));

}

}

**class** RabbitPair{

**int** age = 0;

**public** **void** increaseAgeByAMonth(){

age++;

}

**public** **void** setAge(**int** age){

**this**.age = age;

}

**public** **int** getAge(){

**return** age;

}

**public** **boolean** isMature(){

**return** (age >= 3) ? **true** : **false**;

}

}