question 1

## **문제**

피보나치 수는 0과 1로 시작한다. 0번째 피보나치 수는 0이고, 1번째 피보나치 수는 1이다. 그 다음 2번째 부터는 바로 앞 두 피보나치 수의 합이 된다.

이를 식으로 써보면 Fn = Fn-1 + Fn-2 (n>=2)가 된다.

n=17일때 까지 피보나치 수를 써보면 다음과 같다.

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597

n이 주어졌을 때, n번째 피보나치 수를 구하는 프로그램을 작성하시오.

## **입력**

첫째 줄에 n이 주어진다. n은 45보다 작거나 같은 자연수이다.

## **출력**

첫째 줄에 n번째 피보나치 수를 출력한다.

answer 1 - good

import java.util.Scanner;

public class T\_2724 {

public static void main(String[] args) {

Scanner scan =new Scanner(System.in);

int count = scan.nextInt();

int[] array={0,1,1};

for(int i=1;i<count;i++){

array[2]=array[0]+array[1];

array[0]=array[1];

array[1]=array[2];

}

if(count==0){

System.out.println("0");

}else{

System.out.println(array[2]);

}

}

}

question 2

## **문제**

자연수 N 과 정수 K 가 주어졌을 때 이항 계수 (NK) 를 구하는 프로그램을 작성하시오.

## **입력**

첫째 줄에 N 과 K 가 주어진다. (1 ≤ N ≤ 10, 0 ≤ K ≤ N )

## **출력**

(NK) 를 출력한다.

answer 1 - runtime error // 범위 착오 에러

import java.util.Scanner;

public class T\_11050 {

static int[][] nk = new int[5][5];

public static void main(String[] args) {

Scanner scan =new Scanner(System.in);

int n=scan.nextInt();

int k=scan.nextInt();

System.out.println(cal(n-1,k-1));

}

// nCr = n-1Cr-1 + n-1Cr

public static int cal(int n,int k){

if(n==k ||k==-1){

return 1;

}else if(nk[n][k]>0){

return nk[n][k];

}

return nk[n][k]=cal(n-1,k-1)+cal(n-1,k);

}

}

answer 2 - good

import java.util.Scanner;

public class T\_11050 {

static int[][] nk = new int[10][10];

public static void main(String[] args) {

Scanner scan =new Scanner(System.in);

int n=scan.nextInt();

int k=scan.nextInt();

System.out.println(cal(n-1,k-1));

}

// nCr = n-1Cr-1 + n-1Cr

public static int cal(int n,int k){

if(n==k ||k==-1){

return 1;

}else if(nk[n][k]>0){

return nk[n][k];

}

return nk[n][k]=cal(n-1,k-1)+cal(n-1,k);

}

}

question 3

## **문제**

다음 소스는 N번째 피보나치 함수를 구하는 함수이다.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | int fibonacci(int n) {  if (n==0) {  printf("0");  return 0;  } else if (n==1) {  printf("1");  return 1;  } else {  return fibonacci(n‐1) + fibonacci(n‐2);  }  } |

fibonacci(3)을 호출하면 다음과 같은 일이 일어난다.

fibonacci(3)은 fibonacci(2)와 fibonacci(1) (첫 번째 호출)을 호출한다.

fibonacci(2)는 fibonacci(1) (두 번째 호출)과 fibonacci(0)을 호출한다.

두 번째 호출한 fibonacci(1)은 1을 출력하고 1을 리턴한다.

fibonacci(0)은 0을 출력하고, 0을 리턴한다.

fibonacci(2)는 fibonacci(1)과 fibonacci(0)의 결과를 얻고, 1을 리턴한다.

첫 번째 호출한 fibonacci(1)은 1을 출력하고, 1을 리턴한다.

fibonacci(3)은 fibonacci(2)와 fibonacci(1)의 결과를 얻고, 2를 리턴한다.

이 때, 1은 2번 출력되고, 0은 1번 출력된다. N이 주어졌을 때, fibonacci(N)을 호출했을 때, 0과 1이 각각 몇 번 출력되는지 구하는 프로그램을 작성하시오.

## **입력**

첫째 줄에 테스트 케이스의 개수 T가 주어진다. 각 테스트 케이스는 다음과 같이 구성되어있다.

첫째 줄에 N이 주어진다. N은 40보다 작거나 같은 자연수 또는 0이다.

## **출력**

각 테스트 케이스마다 0이 출력되는 횟수와 1이 출력되는 횟수를 공백으로 구분해서 출력한다.

answer 1 - good

import java.util.Scanner;

public class T\_1003 {

static int[][] count=new int[9999][2];

public static void main(String[] args) {

Scanner scan=new Scanner(System.in);

int number=scan.nextInt();

int[] inputs=new int[number];

for(int i=0;i<number;i++){

inputs[i]=scan.nextInt();

if(count[i][0]!=0){

continue;

}

count[inputs[i]][0]=cal(inputs[i],0);

count[inputs[i]][1]=cal(inputs[i],1);

}

for(int i=0;i<number;i++){

System.out.println(count[inputs[i]][0]+" "+count[inputs[i]][1]);

}

}

public static int cal(int input,int check){

if(input==0){

if(check==1){

return 0;

}else{

return 1;

}

}else if(input==1){

if(check==1){

return 1;

}else{

return 0;

}

}

return count[input][check] = cal(input-1,check)+cal(input-2,check);

}

}

question 4

## **문제**

정수 N개로 이루어진 수열 A와 정수 X가 주어진다. 이 때, A에서 X보다 작은 수를 모두 출력하는 프로그램을 작성하시오.

## **입력**

첫째 줄에 N과 X가 주어진다. (1 ≤ N, X ≤ 10,000)

둘째 줄에 수열 A를 이루는 정수 N개가 주어진다. 주어지는 정수는 모두 1보다 크거나 같고, 10,000보다 작거나 같은 정수이다.

## **출력**

X보다 작은 수를 입력받은 순서대로 공백으로 구분해 출력한다. X보다 작은 수는 적어도 하나 존재한다.

answer 1 - good

import java.util.Scanner;

public class T\_10871 {

public static void main(String[] args) {

Scanner scan= new Scanner(System.in);

int count=0,size=scan.nextInt();

int[] N=new int[size];

int x=scan.nextInt();

for(int i=0;i<size;i++){

int check=scan.nextInt();

if(check<x){

N[count]=check;

count++;

}

}

System.out.print(N[0]);

for(int i=1;i<count;i++){

System.out.print(" "+N[i]);

}

}

}