Application of Domino's Tiling Theory to calculate the probability of tatami mat arrangements in rectangular rooms, the number of mats required, and the areas where a full sheet of tatami mat cannot be arranged

Institute: Samsenwittayalai School | Thailand(TH) **Students:** Saran Laiwisetkul, Saknarin Srima, Tinnapat Sittisuwan **Supervisor :** Mr. Sutthiphan Hiranpruk

Abstract

In Japanese folklore, mats with a ratio of 1:2 are popular. The mat is called Tatami. After the organizers have studied Tatami, the arrangement of tiles in one room can be arranged in many different ways. This made the organizers interested in counting the number of possible ways to arrange the tiles in one room. So we built a computer program that can calculate the things we care about. There are 3 items as follows: 1. The number of methods for arranging the Tatami tiles into the room size 2*n and 3*n. 2. The total number of tiles used in the arrangement and 3. The remaining room space after the arrangement. tile By using knowledge of Domino's Tiling Theorem, Recursive Function, Fibonacci sequence, and computer programming. therefore became “Application of Domino's Tiling Theory to calculate the probability of tatami mat arrangements in rectangular rooms, the number of mats required, and the areas where a full sheet of tatami mat cannot be arranged”. The results showed that the programming output showed consistent results for all three experiments: a comparison with the results of a mathematical formula according to Domino's tiling Theorem, a comparison with the results of a mathematical formula of the form. General and comparison with the results of the distribution of all possible number of methods. And part of the program can be used or extended in industrial applications, the cost of tatami mats according to the amount that can be placed in 2*n and 3*n rooms, is used as a reference for work that brings Knowledge of mathematics and computers, as well as being able to apply data and program formats to further developments.

Introduction

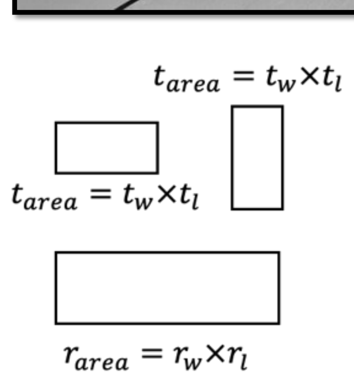
Dominoes , the well known board game. It's a tile-based board game that played with the two-sided rectangular domino pieces. the side of the domino piece divided into to two square end. Each end contain the numbers of dots (pips) from 0 to 6 dots. A set of domino contain 28 domino pieces, sometimes called a deck or a pack. How to play dominoes is to blocking them by matching the domino piece that have the same number of dots. We found that the play of the dominoes resemble the process of placing Tatami mats of the Japanese. Then we use the method of Domino Tiling to solve the process of placing the Tatami mats.

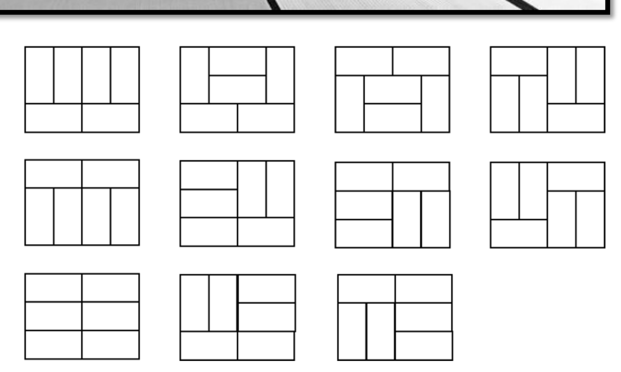
Objectives

We are interested in studying the number of possibilities for creating a tatami mat. By arranging according to the Domino tiling theory, we observe that we can apply the Tatami tiling pattern to the theory through coding and building it into an application

Tatami program

```c
#include <stdio.h>
#include <math.h>
#include <string.h>

int r_area, t_area;
int r_w, r_l, t_w, t_l;
int x=1;
void menu(void);

int tatami_num(int r_w, int r_l, int t_w, int t_l){
    r_area = r_w*r_l;
    t_area = t_l*t_w;

    if(t_w>r_w||t_w>r_l||t_l>r_w||t_l>r_l){
        printf("Cant fit");
        return 0;
    }
    else return(floor(r_area/t_area));
}

float rest_area(int r_w, int r_l, int t_w, int t_l){
    return (r_area - t_area*floor(r_area/t_area));
}

int twobyn(int n){
    if (n <= 2)
      return n;
    return twobyn(n - 1) + twobyn(n - 2);
}

int threebyn(int n){
    int A[n + 1], B[n + 1];
    A[0] = 1, A[1] = 0, B[0] = 0, B[1] = 1;
    for (int i = 2; i <= n; i++) {
        A[i] = A[i - 2] + 2 * B[i - 1];
        B[i] = A[i - 1] + B[i - 2];
    }
    return A[n];
}

int main(){
    menu();
    if(r_w == 2){
        printf("No. of Solution : %d\n", twobyn(r_l));
    }
    if(r_w == 3){
        printf("No. of Solution : %d\n", threebyn(r_l));
    }
    printf("The number of No. of Tatami : %d\n", tatami_num(r_w,r_l,t_w,t_l));
    printf("The number of Rest area in the room : %.2f square meters\n",
rest_area(r_w,r_l,t_w,t_l));
    return 0;
}

void menu(void){
    int choice;
    printf("——— Welcome to the tatami Calculator ———\n");
    printf("Enter the room width (2 or 3) : ");
    scanf("%d", &r_w);
    printf("Enter the room length : ");
    scanf("%d", &r_l);
    printf("**--**--**--**--**\nEnter number\n1: Tatami Original Size\n2: Custom
Size\n**--**--**--**--**\n");
    scanf("%d", &choice);
    if(choice == 1){
        t_w = 1;
        t_l = 2;
    }
    if(choice == 2){
        printf("Enter the Tatami width : ");
        scanf("%d", &t_w);
        printf("Enter the Tatami width : ");
        scanf("%d", &t_l);
    }
}
```

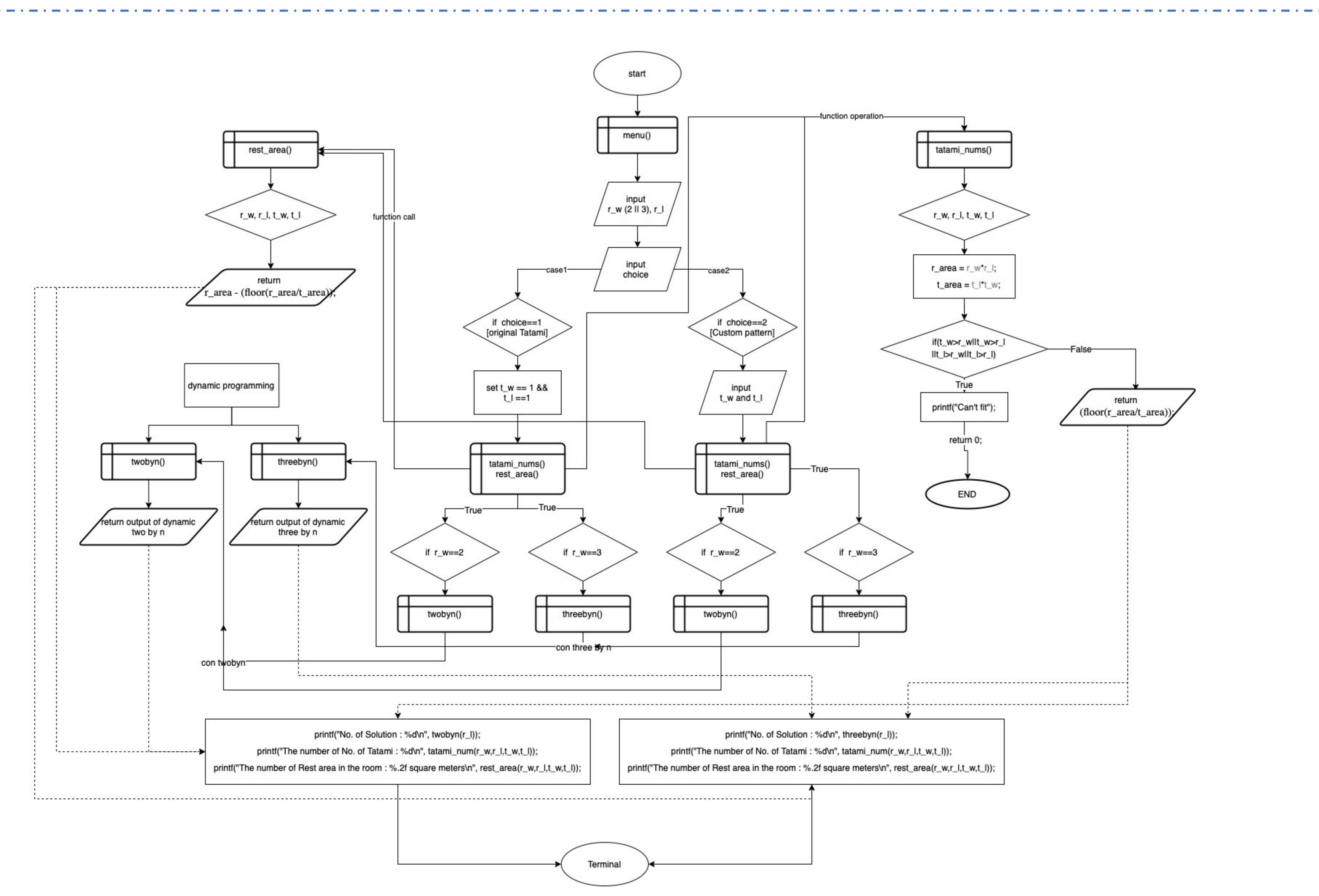




Design the variables and formula

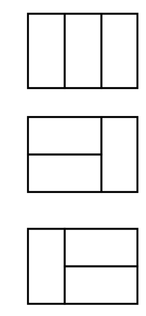
There are four input variables: *r_w (room width), r_l (room length), t_w (Tatami width) and t_l (Tatami length).* The secondary variable s is *r_area* and *t_area*. Which can find the area according to the equation $r_{area} = r_{width} \times r_{length}$ and $t_{area} = t_{width} \times t_{length}$. You can use scanf() to get the values and printf(). Show values at the end of the program.

build program accordingly Flowchart of Tatami Calculator Programs



Test the result

```
------- Welcome to the tatami Calculator -------
Enter the room width (2 or 3) : 2
Enter the room length : 3
**--**--**--**--**
Enter number
1: Tatami Original Size
2: Custom Size
**--**--**--**--**
1
No. of Solution : 3
The number of No. of Tatami : 3
The number of Rest area in the room : 0.00 square meters
```

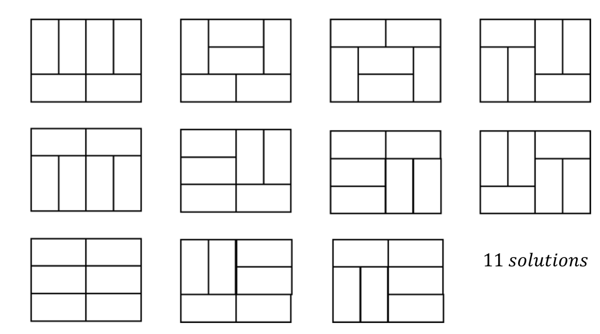


3 *solutions*

$$f_n = f_{n-1} + f_{n-2} \text{ when } f_1 = 1, f_2 = 2$$

From the fibonucci number and domino’s tiling formula is $f_n = f_{n-1} + f_{n-2}$ when $f_1 = 1, f_2 = 2$ In this case the input(n) is 3 so $f_n = f_{n-1} + f_{n-2}$ is $f_3 = f_2 + f_1 = 1 + 2 = 3\ solutions.$

```
------- Welcome to the tatami Calculator -------
Enter the room width (2 or 3) : 3
Enter the room length : 4
**--**--**--**--**
Enter number
1: Tatami Original Size
2: Custom Size
**--**--**--**--**
1
No. of Solution : 11
The number of No. of Tatami : 6
The number of Rest area in the room : 0.00 square meters
```



11 *solutions*

```
------- Welcome to the tatami Calculator -------
Enter the room width (2 or 3) : 2
Enter the room length : 2
**--**--**--**--**
Enter number
1: Tatami Original Size
2: Custom Size
**--**--**--**--**
1
No. of Solution : 2
The number of No. of Tatami : 2
The number of Rest area in the room : 0.00 square meters
```

$$A_n = 2^{2n^2} \prod_{i=1}^{n} \prod_{j=1}^{n} \left[\cos^2\left(\frac{i\pi}{2n+1}\right) + \cos^2\left(\frac{j\pi}{2n+1}\right)\right]$$

When in the form of $2n \times 2n$. So in this case n=1. Then the value n=1 is substituted into the equation. After substituting the values into the equation, it was found that $A_n = 2^2 \prod_{i=1}^{1} \prod_{j=1}^{1} \left[\cos^2\left(\frac{\pi}{3}\right) + \cos^2\left(\frac{\pi}{3}\right)\right]$. $\prod_{i=1}^{n} \prod_{j=1}^{n} k$, It can have a value of k because it is a sum from n=1 to n=1. Then $A_n = 2^2 \left[\cos^2\left(\frac{\pi}{3}\right) + \cos^2\left(\frac{\pi}{3}\right)\right]$. So $A_n = 2^2 \left[\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2\right]$.

$A_n = 2^2 \left[\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2\right] = 2^2 \left[\frac{1}{2}\right] = 2\ solutions.$

DOMINO’S TILING THEOREM

Domino tiling of space in the Euclidean plane is the tessellation of space by dominoes [1×2]. A shape formed by the merging of two squares converging edge-to-edge. in the same way It is a perfect match in a point graph formed by placing the vertex in the center of each square of the area. and connect two vertices when they match adjacent squares. and can find the number of arrangements according to the following formula.

$$A_n = 2^{2n^2} \prod_{i=1}^{n} \prod_{j=1}^{n} \left[\cos^2\left(\frac{i\pi}{2n+1}\right) + \cos^2\left(\frac{j\pi}{2n+1}\right)\right]$$

$$A_{m,n} = \prod_{j=1}^{m/2} \prod_{k=1}^{n/2} \left[4\cos^2\left(\frac{j\pi}{m+1}\right) + 4\cos^2\left(\frac{k\pi}{n+1}\right)\right]$$

Mathematical knowledge

Floor function and Ceiling function, Recursive function, Summation and double summation $\prod_j^m \prod_k^n$, Basic probability, Basic Fibonacci, Basic Trigonometry, Exponential function, Basic Function.

Computational knowledge

Basic of C programming, nested loop, for loop, Extended math library(math.h), Extended math string (string.h). Fibonacci and recursive function, c Function, application flowchart, if-else condition.

Conclusion

The result of the program for counting the number of tiling methods is equal to the calculation using domino tiling theorems.

$$A_n = 2^{2n^2} \prod_{i=1}^{n} \prod_{j=1}^{n} \left[\cos^2\left(\frac{i\pi}{2n+1}\right) + \cos^2\left(\frac{j\pi}{2n+1}\right)\right]$$
$$T^4_{(m,n)} = \prod_{i=1}^{n} \prod_{j=1}^{m} 4\left[\cos^2\left(\frac{i\pi}{n+1}\right) + \cos^2\left(\frac{j\pi}{m+1}\right)\right]$$

and list all probability counts of patterns. And we plotted the graph