

# 課題 1 : ヒストグラムの生成

$a\sim z$ ,  $A\sim Z$ ,  $0\sim 9$  で構成された文字列 `str` が与えられたとき、それぞれの文字が何回出現するかを数えたヒストグラムを返す関数を作れ。答えは長さ  $26 + 26 + 10 = 62$  の整数の配列とする。英小文字、英大文字、数字の順に出現回数をいれて返すこと。

作成する関数 : `kadai1(str)`

`str = "yaaaamamoto998meijo"` の場合、

`kadai1(str) =`  
[ 5, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 3, 0, 3, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,  
1, 0,  
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 2 ]

この下線部が小文字 a-z に対応

この下線部が小文字 a-z に対応

「m」が3回出現している

「9」が2回出現している

ただし、文字列 `s` の `i` 番目の文字を取り出した `i` 文字の文字列は、`s.charAt(i)`、その文字コードは `s.charCodeAt(i)` で計算できる

# 課題2：第2種スターリング数

組み合わせ数に似た数として第2種スターリング数がある。まず、この数を計算する関数 `stirling(n, k)` を計算する関数をつくり、つぎにこの関数によって定義されるベル数  $B(n)$  を計算する関数 `bell(n)` を作れ。課題の提出はいかに示すようにせよ。

$${}_n S_1 = {}_n S_n = 1$$

$${}_n S_k = {}_{n-1} S_{k-1} + k \cdot {}_{n-1} S_k$$

$$B(n) = \sum_{k=1}^n {}_n S_k$$

ベル数は右の表のような値となる。

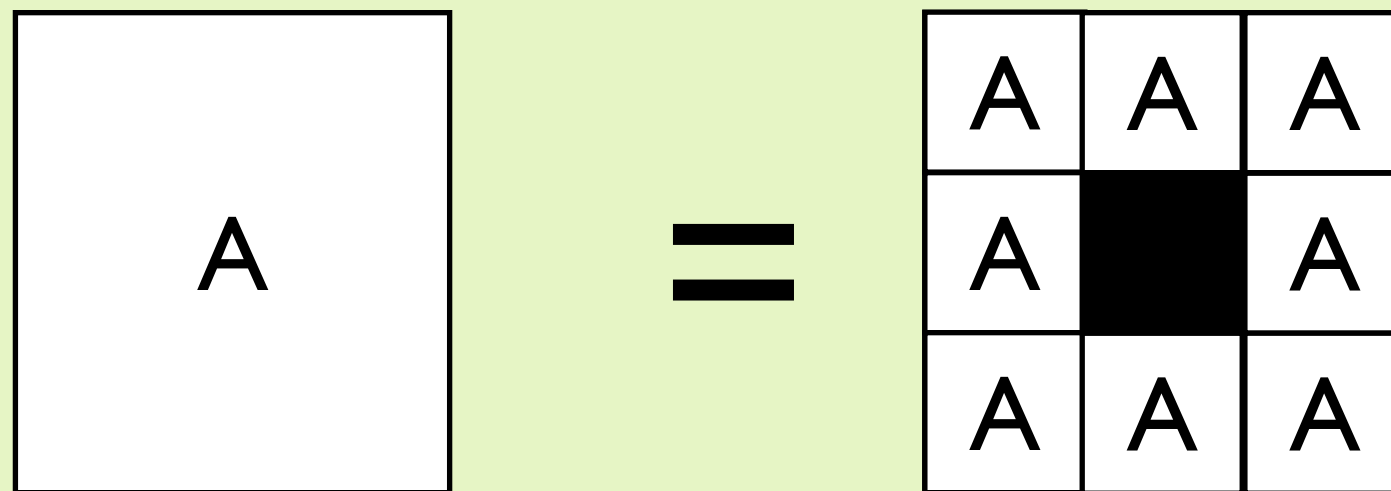
1	1	
2	2	
3	5	<b>Bell数</b>
4	15	
5	52	
6	203	
7	877	
8	4140	
9	21147	
10	115975	
11	678570	
12	4213597	
13	27644437	
14	190899322	
15	1382958545	
16	10480142147	
17	82864869804	
18	682076806159	
19	5832742205057	

提出は、関数 `a = stirling(n, k)` と `b = bell(n)` を両方計算して配列にして返す `kadai2(n, k) = [a, b]` を提出すること。この関数のみを提出すること。

### 課題 3 : シェルピンスキーカーペットの描画(1)

3

以下のような再帰構造でシェルピンスキーガスケットと同じような図形をフラクタル図形として描画するための関数書け。



上記のような性質をもつ図形はどのようなものか？

黒で塗りつぶした長方形を描画するには左下の点と右上の点をそれぞれ `pt1`, `pt2` として, `draw_rect(pt1, pt2)` を呼べば良い. 上図の左のすべて白の図を0次近似として, `n`次近似を描画するための四角形 (この場合には正方形) の左下と右上の座標を順にいた配列を返す関数 `kadai3(n)` を計算するプログラムを書け

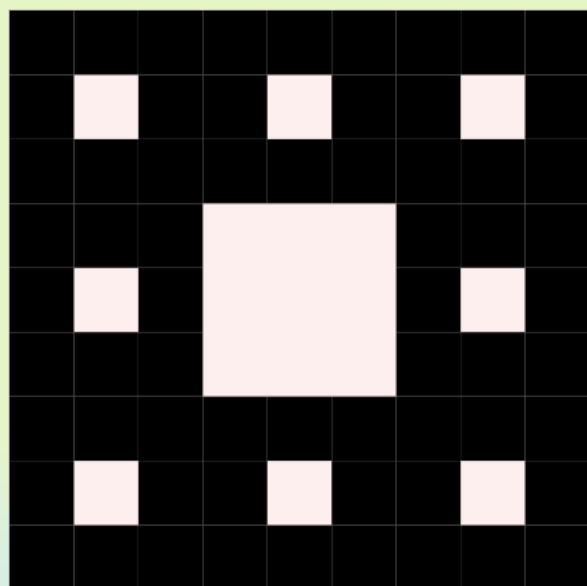
# 課題 3 : シェルピンスキーカーペットの描画(2)

kadai3(n)が正しく出来ていれば,

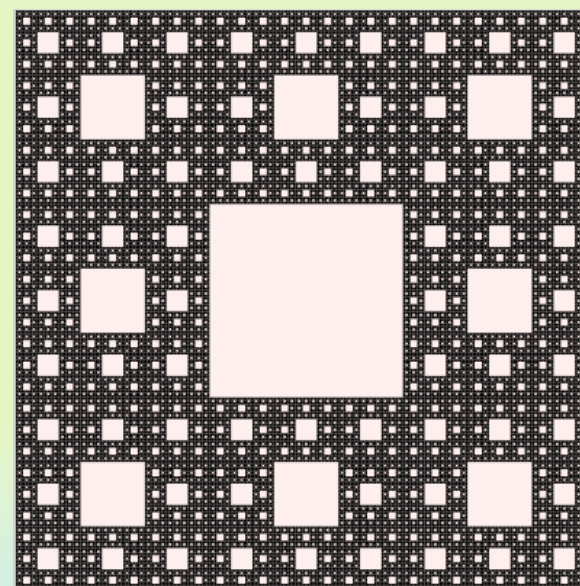
```
n = 5  
ans = kadai3(n)  
for (var i = 0; i < ans.length; i++){  
    var pt1 = ans[i][0]  
    var pt2 = ans[i][1]  
    draw_rect(pt1, pt2)  
}
```

というプログラムを描画環境つき実行環境で動かせば以下のようなになる。 描画環境では左下が(0, 0), 右上が(1, 1)になっている。

描画のた  
めの関数



n = 2



n = 5

 課題 3 : シェルピンスキーカーペットの描画(3) 

## 提出の指針 1 (あるやり方)

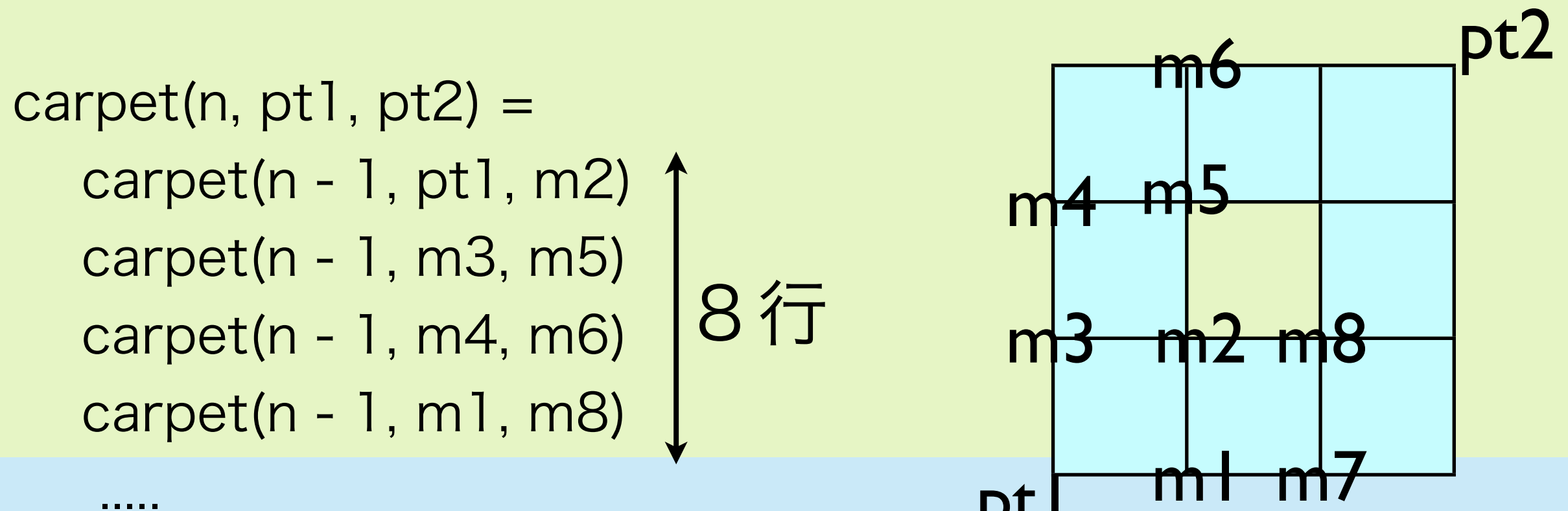
関数 `carpet(n, s)` は大きさ  $s \times s$  の四角形中に描かれる深さ  $n$  のシェルピンスキーカーペットの正方形の座標の組の列を返すとする。このとき、以下のように再帰的に定義できる。

```
carpet(n, s) =  
  if (n == 0) [[(0, 0), (s, s)]]を返す  
  else {  
    var cx = carpet(n - 1, s / 3);  
    cxに含まれる座標の組を8通りに平行移動させて  
    作ることのできる組を集めて dx とする;  
    return dx;  
  }
```

## 提出の指針 2 (別のやり方)

- 授業中に示したSGと同様に考える. あるpt1とpt2が左下, 右上の座標になる正方形領域にn次カーペットを描画する関数を  $\text{carpet}(n, \text{pt1}, \text{pt2})$  とおけば,

$$\text{carpet}(0, \text{pt1}, \text{pt2}) = \text{draw\_rect}(\text{pt1}, \text{pt2})$$



- と考えることができる.