

第二类斯特林数

n 个不同的物品 放进 m 个相同的箱子，每个箱子至少有一个物品 求方案数

递推版：

```
11 f(int n, int m)
{
    if (m <= 0 || n < m)
        return 0;
    if (n == m)
        return 1;
    else
        return f(n-1, m-1) + f(n-1, m) * m;
}
```

打表：

第一类：

```
s[0][0]=1;
for(int i=1;i<=n;i++)
    for(int j=1;j<=i;j++)
        s[i][j]=s[i-1][j-1]+(i-1)*s[i-1][j];
```

第二类：

```
s[0][0]=1;
for(int i=1;i<=n;i++)
    for(int j=1;j<=i;j++)
        s[i][j]=s[i-1][j-1]+j*s[i-1][j];
```

高斯消元法

```
void solve()
{
    int n;
    scanf("%d", &n);
    vector<vector<db>> A(n, vector<db>(n + 1));
    for (int i = 0; i < n; i++)
        for (int j = 0; j <= n; j++)
            scanf("%lf", &A[i][j]);
    for (int i = 0; i < n; i++)//处理每一个未知数
    {
        int index;
        db ma = 0;
        for (int j = i; j < n; j++)
```

```

{
    if (ma < abs(A[j][i]))
    {
        ma = abs(A[j][i]);
        index = j;
    }
}
if (ma == 0)
{
    printf("No Solution\n");
    exit(0);
}
swap(A[i], A[index]);
index = i;
db tmp = A[index][i];
for (int j = i; j <= n; j++)
{
    A[index][j] /= tmp;
}
for (int j = 0; j < n; j++)
{
    if (j == index) continue;
    tmp = A[j][i];
    for (int k = i; k <= n; k++)
    {
        A[j][k] -= tmp * A[index][k];
    }
}
}
for (int i = 0; i < n; i++)
    printf("%.21f\n", A[i][n]);
}

```