

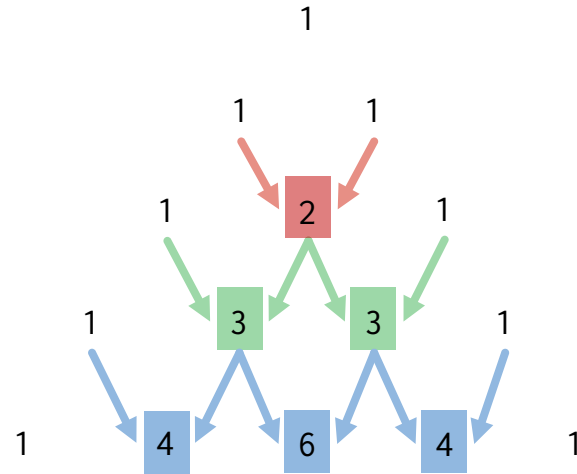
課題4(1)

【チャレンジ問題】

- step1: 段ごとに配列を用意し, 規則性を反映して値を設定
 - どの段も左端と右端の値は1
 - 3段目以降の段では, 左端と右端以外の値はひとつ前の段の左右の値の和

```
int l1[1], l2[2], l3[3], l4[4], l5[5];
int i, j;

l1[0] = 1;
l2[0] = l2[1] = 1;
l3[0] = l3[2] = 1;
for (j = 1; j > 0; j--) {
    l3[j] = l2[j] + l2[j - 1];
}
l4[0] = l4[3] = 1;
for (j = 2; j > 0; j--) {
    l4[j] = l3[j] + l3[j - 1];
}
l5[0] = l5[4] = 1;
for (j = 3; j > 0; j--) {
    l5[j] = l4[j] + l4[j - 1];
}
```



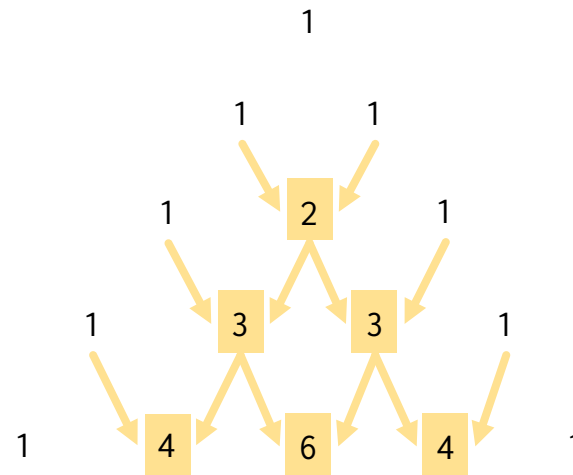
課題4(2)

【チャレンジ問題】

- step2: 1つの配列で複数の段を表現できるよう変更
 - for文の中で「各段の値の設定→表示」を繰り返すことで可能

```
int a[5];
int i, j;

for (i = 0; i < 5; i++) {
    a[i] = 1;
    for (j = i - 1; j > 0; j--) {
        a[j] = a[j] + a[j - 1];
    }
    for (j = 0; j <= i; j++) {
        printf("%2d", a[j]);
    }
    printf("\n");
}
```



```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

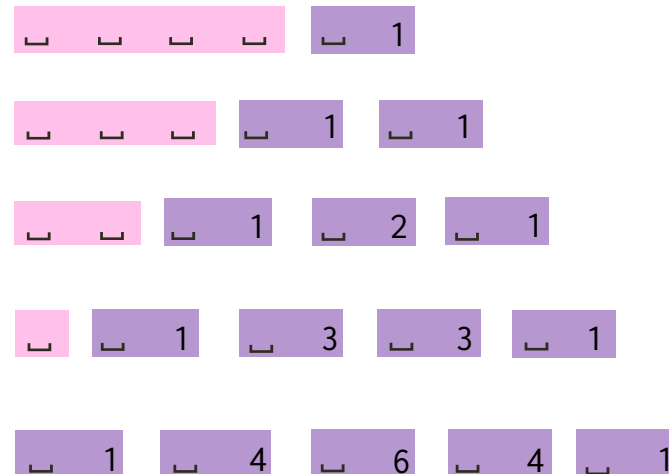
課題4(3)

【チャレンジ問題】

- step3: 各段の値を規則性を反映して表示
 - 各段のすべての値の前には空白がある
 - 画面左端から各段の左端の値が表示されるまでに挿入される空白の数には次の規則性がある
 - 画面左端から左端の値までの空白の数 = 段数 - 段番号 + 1

```
int a[5];
int i, j;

for (i = 0; i < 5; i++) {
    a[i] = 1;
    for (j = i - 1; j > 0; j--) {
        a[j] = a[j] + a[j - 1];
    }
    for (j = 0; j < 5 - i; j++) {
        printf(" ");
    }
    for (j = 0; j <= i; j++) {
        printf("%2d", a[j]);
    }
    printf("\n");
}
```



課題4(4)

【チャレンジ問題】

- step4: 段数はmain()から受け取るようにしてpas()を作成

```
#include <stdio.h>
```

```
void pas(int n) {  
    int a[5], i, j;
```

```
    for (i = 0; i < n; i++) {  
        a[i] = 1;  
        for (j = i - 1; j > 0; j--) {  
            a[j] += a[j - 1];  
        }
```

```
        for (j = 0; j < n - i; j++) {  
            printf(" ");
```

```
        }  
        for (j = 0; j <= i; j++) {  
            printf("%2d", a[j]);  
        }
```

```
        printf("¥n");
```

```
    }
```

```
}
```

(右へ続く)

```
int main() {  
    int n;
```

```
    printf("1以上5以下の整数を入力してください¥n");  
    scanf("%d", &n);  
    pas(n);
```

```
    return 0;  
}
```