

JOIG 2021 A問題 金平糖(Konpeito)

解説: 平木 康傑

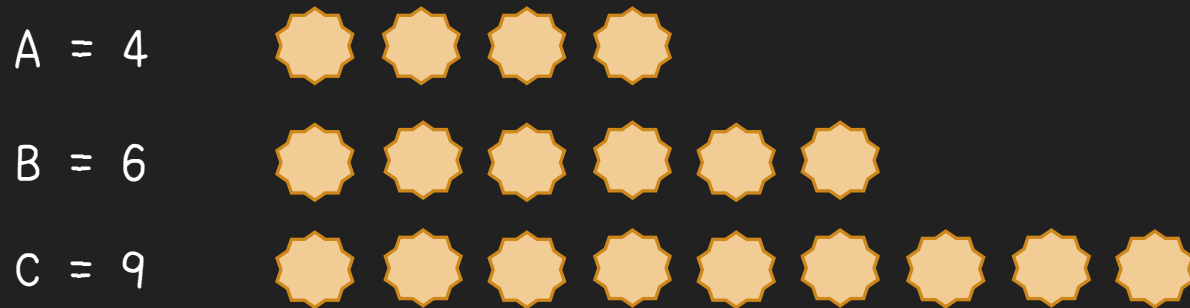


問題概要

- 3人が金平糖を同じ数だけ食べることが目標
- 現時点で3人が食べた個数が A, B, C として与えられる
- 追加で食べるべき個数の合計は最小でいくつ？

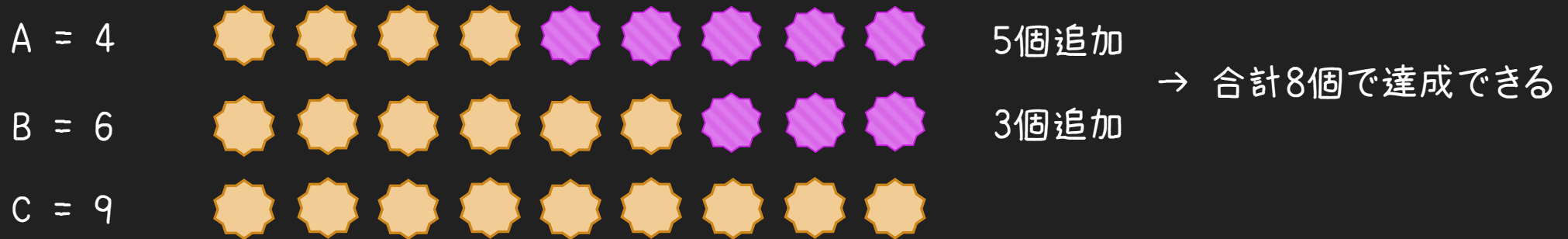
問題概要

- 3人が金平糖を同じ数だけ食べることが目標
- 現時点で3人が食べた個数が A, B, C として与えられる
- 追加で食べるべき個数の合計は最小でいくつ？



問題概要

- 3人が金平糖を同じ数だけ食べることが目標
- 現時点で3人が食べた個数が A, B, C として与えられる
- 追加で食べるべき個数の合計は最小でいくつ？



解法の方針

- A, B, C の値を受け取る(入力する)
- 答えを計算する
- 答えを出力する

解法の方針

- A, B, C の値を受け取る(入力する)
- 答えを計算する
- 答えを出力する

- 言語によって書き方は異なるので、入門書やチュートリアルなどを参考にしてある程度習得しておこう
 - 入力・出力
 - 四則演算など
 - 条件分岐

入力・出力

○ C++

- C++標準で用意された関数が用途ごとに別の「ライブラリ」に詰め込まれている
- 入力・出力関連のライブラリ関数は<iostream>ライブラリに入っている
 - #include <iostream> と書くことで取り込める
- C++の入出力は、標準入出力と結びついた「ストリーム」と演算子とを使って行う
 - std::cin は標準入力, std::cout は標準出力
 - 変数 a, b, c に入力: `std::cin >> a >> b >> c;`
 - 変数 result と改行を出力: `std::cout << result << '\n';`

入力・出力

○ Python

○ Pythonでははじめから組み込み関数が見える

○ 整数 a, b, c を入力: `a, b, c = map(int, input().split())`

○ `input()` で入力を文字列として受け取り, `.split()` で空白ごとに分割して配列にする

○ `map` 関数によって, その配列の各要素に `int` 関数を適用する

○ Python特有の書き方として, たとえば `a, b, c = [1, 2, 3]` とすればまとめて代入できる

○ 変数 `result` を 1 行に出力: `print(result)`

解法の方針 (再掲)

- ~~⊖~~ A, B, C の値を受け取る(入力する) ✓
- 答えを計算する
- ~~⊖~~ 答えを出力する ✓

考察

- 3人が金平糖を同じ数だけ食べることが目標
- いくつにすればいいか？

A = 4



B = 6



C = 9

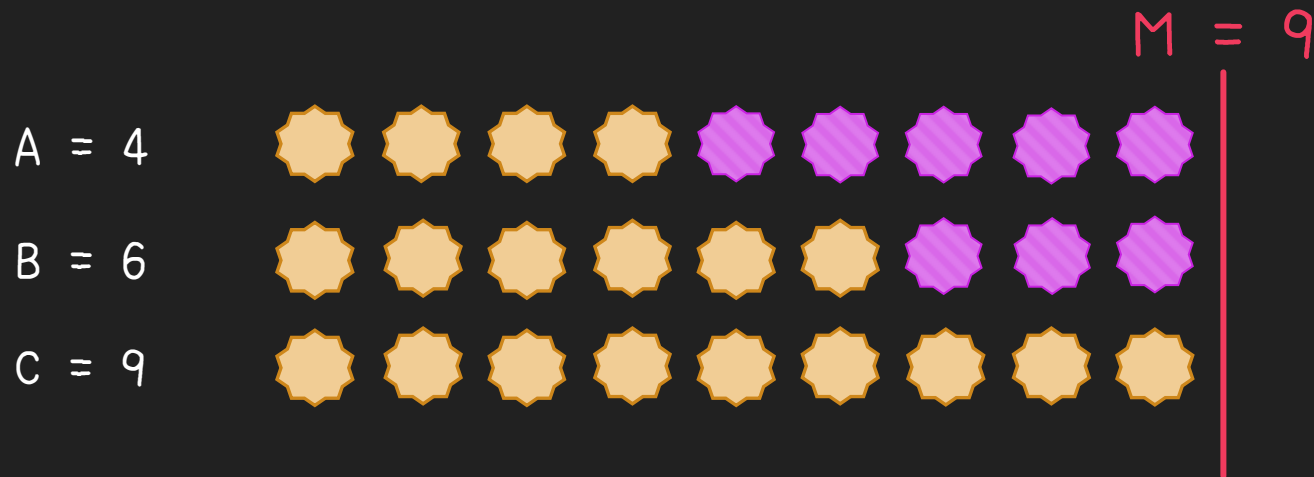


9



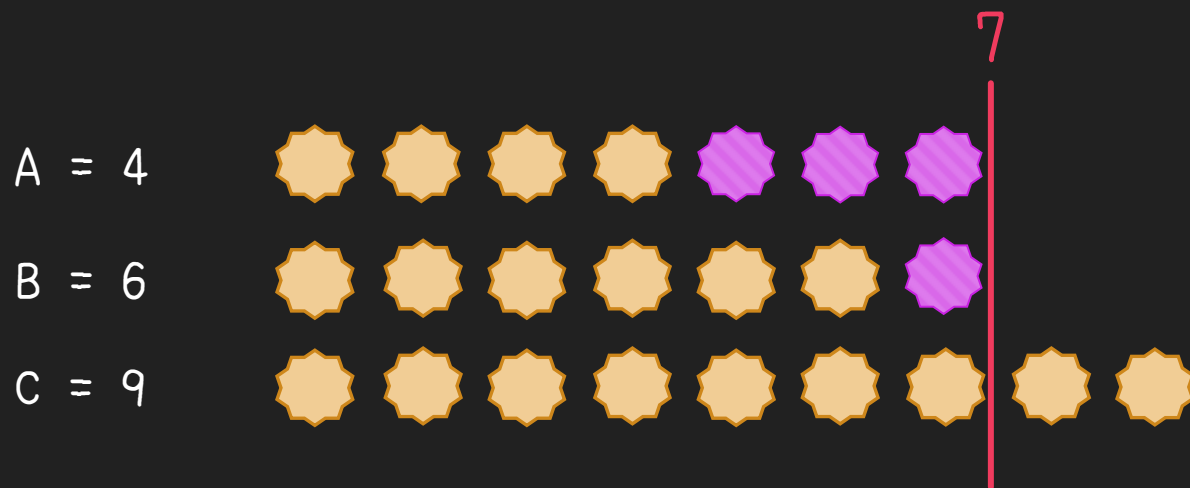
考察

- 3人が金平糖を同じ数だけ食べることが目標
- いくつにすればいいか？
- A, B, C の最大値 (M とする) にすればいい！
- なぜ？



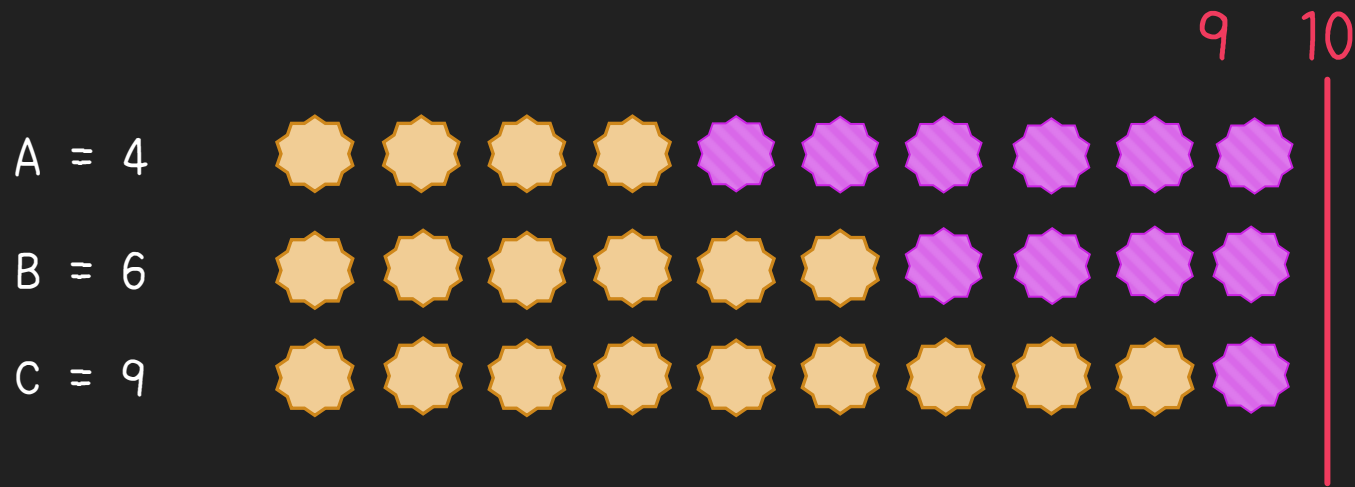
考察 (最大値 M にそろえるべき理由)

- M より小さい個数にそろえることは不可能
 - (もう M 個食べた人がいるから)



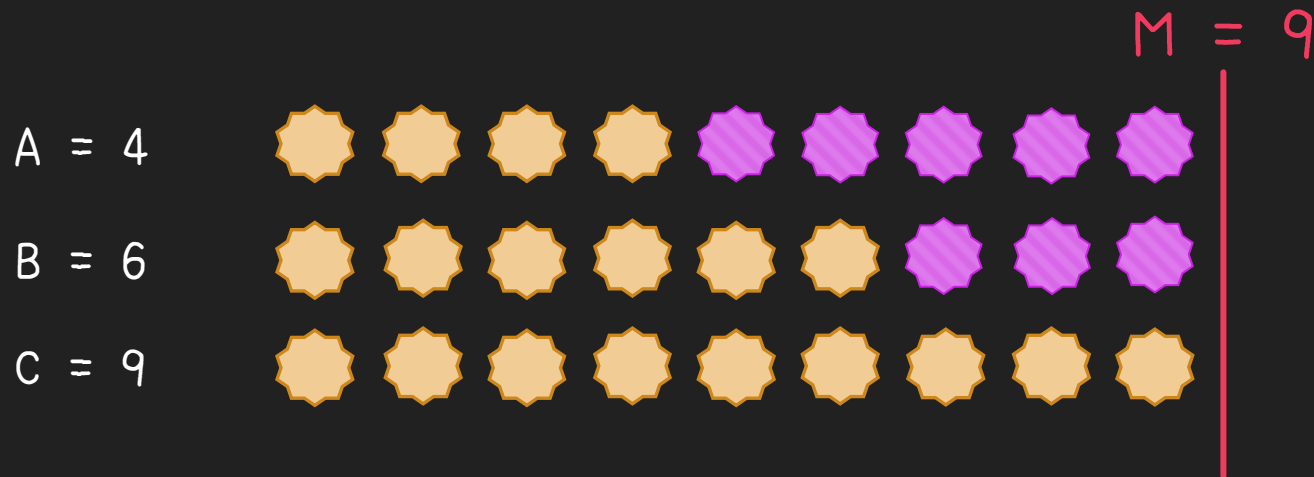
考察 (最大値 M にそろえるべき理由)

- M より大きい個数にそろえるのは無駄がある
 - 全員が追加で1つ以上食べているので, 1ずつ減らして問題ない



考察

- なので, A, B, C のうちの最大値を求めれば, それを M とおくと $(M-A) + (M-B) + (M-C)$ が答えとなる



最大値の求め方

- 小課題 1 では $A < B < C$ が保証されているので、いつでも C が最大値
 - 40点がもらえる
- 小課題 2 ではちゃんと最大値を求める必要がある

最大値の求め方

- 場合分け

- $(A \geq B$ かつ $A \geq C)$ なら A , $(B \geq C$ かつ $B \geq A)$ なら B , どちらでもなければ C

- 順に更新

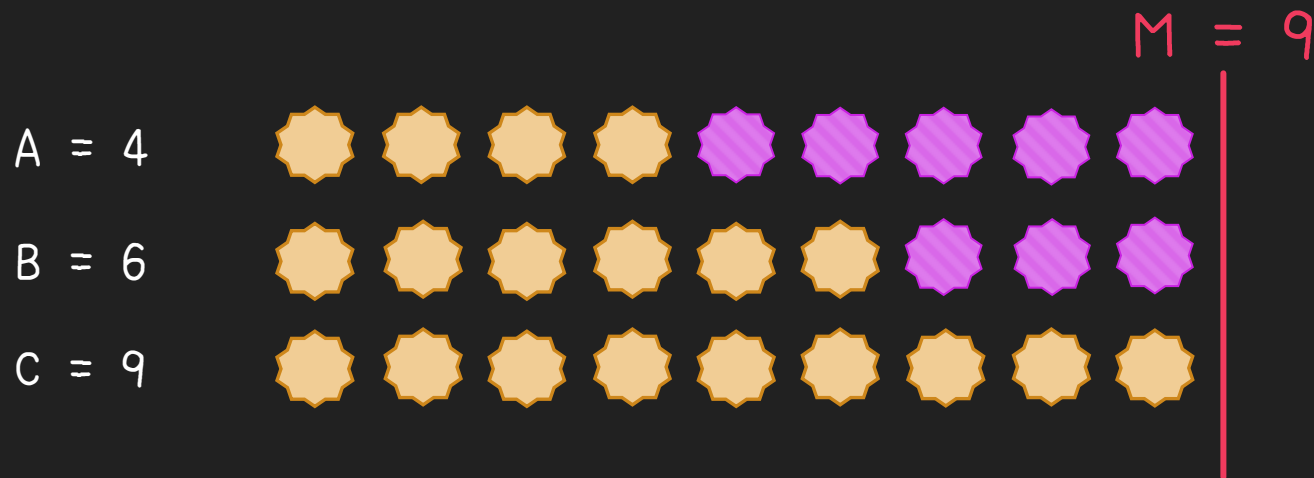
- はじめ $M = 0$ とする

- A, B, C と順番に比べ, M と比較対象のうち大きい方を新たな M にする

- 条件分岐を適切に使おう

考察 (再掲)

- A, B, Cのうちの最大値を求めれば, それを M とおくと $(M-A) + (M-B) + (M-C)$ が答えとなる



解法の方針（再掲）

- ~~⊖ A, B, C の値を受け取る(入力する)~~ ✓
- ~~⊖ 答えを計算する~~ ✓
- ~~⊖ 答えを出力する~~ ✓

実装例

```
#include <iostream>

int main (void) {
    int A, B, C;

    std::cin >> A >> B >> C;

    int maxv = 0;
    if (maxv < A) maxv = A;
    if (maxv < B) maxv = B;
    if (maxv < C) maxv = C;

    int result = 0;
    result += (maxv - A);
    result += (maxv - B);
    result += (maxv - C);

    std::cout << result << '\n';

    return 0;
}
```

C++

```
import java.util.*;
```

```
public class Main {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);

        int a = Integer.parseInt(sc.next());
        int b = Integer.parseInt(sc.next());
        int c = Integer.parseInt(sc.next());

        int maxv = 0;
        if (a >= b && a >= c) {
            maxv = a;
        } else if (b >= c && b >= a) {
            maxv = b;
        } else {
            maxv = c;
        }

        int result = (maxv - a) + (maxv - b) + (maxv - c);

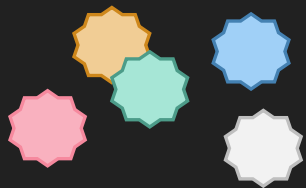
        System.out.println(result);
    }
}
```

Java

得点分布

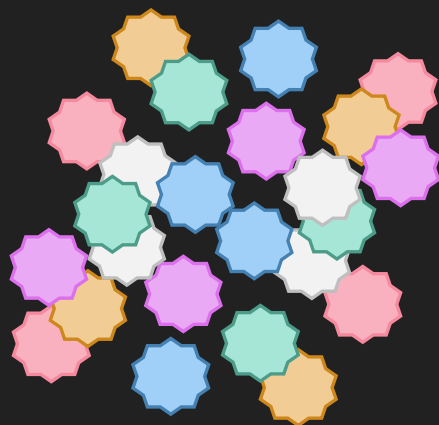
0点

5人



40点

24人



100点

96人

