

演習1 解答

```
void start() {
    String input1 = JOptionPane.showInputDialog("一つ目の数値を入力");
    String input2 = JOptionPane.showInputDialog("二つ目の数値を入力");
    String input3 = JOptionPane
        .showInputDialog("どのような演算をおこないますか？ \n(1) 和\n(2) 差\n(3) 積\n(4) 商");
    double a = Double.parseDouble(input1);
    double b = Double.parseDouble(input2);
    int calcu = Integer.parseInt(input3);
    double result = Calculator(a, b, calcu);
    JOptionPane.showMessageDialog(null, "計算結果は" + result + "です");
}

double Calculator(double a, double b, int calcu) {
    if (calcu == 1) {
        return a + b;
    } else if (calcu == 2) {
        return a - b;
    } else if (calcu == 3) {
        return a * b;
    } else {
        return a / b;
    }
}
}
```

演習2

解答

```
void start() {
    String SizeS = JOptionPane.showInputDialog("配列の長さを指定してください");
    int BoxSize = Integer.parseInt(SizeS);
    int[] numbers = new int[BoxSize];
    for (int i = 0; i < numbers.length; i++) {
        String InputNum = JOptionPane.showInputDialog((i + 1)
            + "番目の数値を入力してください");
        numbers[i] = Integer.parseInt(InputNum);
    }

    String j = JOptionPane.showInputDialog("配列の場所を選択してください");
    int Num = Integer.parseInt(j) - 1;
    int Count = 0;
    for (int k = 0; k < BoxSize; k++) {
        if (numbers[Num] > numbers[k]) {
            Count++;
        }
    }
    if (Count == 0) {
        JOptionPane
            .showMessageDialog(null, (Num + 1) + "番目の値より小さい部分はありません");
    } else {
        JOptionPane.showMessageDialog(null, (Num + 1) + "番目の値より小さい部分は"
            + Count + "箇所あります");
    }
}
}
```

演習3 解答

```
void start() {
    boolean check = true;
    int Max = 10;
    int Count = 0;
    int CN = 0;
    boolean numcheck = true;
    int Mynumber = 0;
    while (check) {
        numcheck = true;
        while (numcheck) {
            // 自分の数値を入力する。1から3までしか足せないののでそうでない時は再入力をさせる。
            MyNumInput(Count, Mynumber, numcheck);

        }
        Count += Mynumber;
        // ゲームを行う場合(自分の番でゲームオーバーにならない場合)
        if (Count < 10) {
            GameMain(CN, Count, Max, check);
        } else {
            JOptionPane.showMessageDialog(null, "あなたの負けです。");
            check = false;
        }
    }
}
```

演習3 解答続き

```
void GameMain(int CN, int Count, int Max, boolean check) {
    CN = CpuNum(Max, Count); // CPUの数値を取得
    Count += CN;
    JOptionPane.showMessageDialog(null, "CPUの入力:" + CN + "\n現在のカウント:"
        + Count);
    if (Count >= 10) {
        JOptionPane.showMessageDialog(null, "あなたの勝ちです。");
        check = false;
    }
}

int CpuNum(int max, int count) {
    int Gap = max - count;
    if (Gap == 3) {
        return 2;
    } else if (Gap <= 2) {
        return 1;
    } else {
        return (int) (2 * (Math.random() + 1));
    }
}
}
```