

問1. Java 言語に関して、次の文章の空欄 (a) ~ (n) を埋めよ。

- あるクラスのオブジェクトをコンストラクト (作成) する場合には、キーワード (a) を使って、そのクラスの (b) を呼び出す。
- あるクラスに属するメソッドを区別する (c) は、メソッドの名前と引数の (d) の並びからなる。また、同じ名前でも引数の (d) の並びが異なるメソッドを作ることを、メソッドの (e) と呼ぶ。
- 自分のオブジェクトのフィールドを明示的に参照したい場合は、キーワード (f) を用いる。
- あるクラス A を元に、拡張・変更した新しいクラス B を宣言することを、クラス A を (g) して、クラス B を宣言すると言う。このとき、クラス A を親クラスあるいは (h) と呼び、クラス B を子クラスあるいは (i) と呼ぶ。
- 親クラスと同じ (c) を持ったメソッドを子クラスで定義することを、メソッドの (j) と呼ぶ。
- 親クラスのメソッドを参照したいときには、キーワード (k) を使う。
- データを昇順あるいは降順に並び替えるソートの方法には、選択ソート、(l)、(m)、マージソートなどがある。データ量が多くなれば、ソートにかかる時間は、選択ソートを使う方がマージソートを使うよりも (n) い。

問2.

次のプログラムに関して、以下の問に答えよ。

```
class Team {
    int point = 0, diff = 0;
    String name;
    Team(String inName) {
        name = new String(inName);
    }
    static void match(Team t1, int p1,
        Team t2, int p2) {
        if (p1 > p2) t1.point += 3;
        else if (p1 == p2) {
            t1.point += 1; t2.point += 1;
        } else t2.point += 3;
        t1.diff += p1 - p2; t2.diff += p2 - p1;
    }
    static boolean gt(Team t1, Team t2) {
        if (t1.point > t2.point) return true;
        if (t1.point < t2.point) (ア)
        if (t1.diff > t2.diff) (イ)
```

```
(ウ)
    }
}
class Yosen {
    public static void main(String[] args) {
        Team aus = new Team("オーストラリア");
        Team bra = new Team("ブラジル");
        Team cro = new Team("クロアチア");
        Team jpn = new Team("日本");
        Team.match(bra, 1, cro, 0);
        Team.match(aus, 3, jpn, 1);
        Team.match(bra, 2, aus, 0);
        Team.match(jpn, 0, cro, 0);
        Team.match(jpn, 1, bra, 4);
        Team.match(cro, 2, aus, 2);
        Team[] grF = new Team[4];
        grF[0] = aus; grF[1] = bra;
        grF[2] = cro; grF[3] = jpn;
        for(int i = 0 ; i < grF.(エ); i++) {
            int maxj = i;
            for(int j = (オ) ;
                j < grF.(カ) ; j++) {
                if ( (キ) ) {
                    maxj = j;
                }
            }
            Team tmp = grF[maxj];
            (ク)
            (ケ)
        }
        System.out.println(grF[0].name
            + "が1番で予戦通過。");
        System.out.println(grF[3].name
            + "がビリ。また来れるといいですね。");
    }
}
```

- チーム (Team) の評価は、point が大きい方が高く、point が同じ場合は diff が大きい方が高い。Team のクラスメソッド gt は、Team のオブジェクトの t1 と t2 を引数として、t1 の方が評価が高いときは true を、そうでないときは false を返すものである (評価が等しいときも false を返す)。 (ア) から (ウ) の空欄を埋めて、このメソッドを完成させなさい。
- クラス Yosen の main メソッドの for 文以下で、(1) の大小関係に従って、降順に選択ソートを行なっている (チームのオブジェクトの評価が等しい場合の結果の順番は不定で良い)。できるだけ実行時に無駄がないように、また、領域が確保された配列の長さ対しては length を使って、(エ) から (ケ) の空欄を埋めて、このメソッドを完成させなさい。
- このプログラムの実行結果として出力されるものを、できるだけ出力される通りに書きなさい。

得失点差が累積していないという指摘により、修正してあります。問題とは関係ないところですので、成績 (サッカーの方ではなく学業の) などには影響しません。

解答

問1.

- (a) `new`
- (b) コンストラクタ
- (c) シグネチャー
- (d) 型
- (e) オーバーロード
- (f) `this`
- (g) 継承
- (h) スーパークラス
- (i) サブクラス
- (j) オーバーライド
- (k) `super`
- (l) ,(m) バブルソート, クイックソート, (ヒープソート, バカソート, B-Tree を使ったソートなど)
- (n) 長い (遅い).

問2.

(1)

- (ア) `return false;`
- (イ) `return true;`
- (ウ) `return false;`

(2)

- (エ) `return false;`
- (エ) `length - 1`
- (オ) `i + 1`
- (カ) `length`
- (キ) `Team.gt(grF[j], grF[maxj])`
- (ク) `grF[maxj] = grF[i];`
- (ケ) `grF[i] = tmp;`

(3)

ブラジルが1番で予戦通過.

日本がビリ. また来れるといいですね.