

問 1 . Java 言語とアルゴリズムに関する次の文章の空欄 (a) ~ (D) を埋めよ。

- 16bit の整数型の配列変数 `iList` を宣言し、その要素 5 個分の領域を確保し、その参照情報を `iList` に代入するためには、(a) と記述する。このとき、配列 `iList` のインデックスとして、(b) から (c) までを使うことが可能である。また、配列変数 `iList` の宣言、領域確保と、要素の値を順番に 3, 2, 3, 4, 8 で初期化することを同時に行うためには、(d) と記述する。
- メソッドを区別する (e) は、メソッドの (f) と (g) の (h) の並びからなる。一つのクラスの中で、(f) が同じで、異なる (g) の (h) の並びをもつメソッドを定義することを、メソッドの (i) と言う。
- 変数のスコープとは、(j) である。識別子 `v` が、フィールドとローカル変数の両方で宣言されていて、両方のスコープに含まれる場合、フィールドの方を参照したい場合は (k) と記述する。
- (l) は、オブジェクトを作成する特別なでありメソッドである。クラス `Student` のクラス型変数 `std1` を宣言し、デフォルトの (l) でオブジェクトを生成し、その参照情報を `std1` に代入するためには、(m) と記述する。また、`std1` のフィールド `pPhy` に 123 を代入するときには、(n) と記述する。自分のクラスの (g) の (h) の並びが異なる (l) を呼び出すときには、キーワード (o) を用いる。
- クラス `Student` を元に、拡張・変更した新しいクラス `UnivStudent` を宣言することを、`Student` を (p) (漢字 2 文字) して、`UnivStudent` を宣言すると言う。このとき、クラス (q) を親クラスまたは (r) クラス、クラス (s) を subclasses または (t) クラスと呼ぶ。子クラスのコンストラクタから親クラスのコンストラクタを呼び出すときには、キーワード (u) を用いる ((q), (s) には `Student` または `UnivStudent` が入る)。
- 親クラスにあるメソッドと同じ (e) を持つメソッドを、子クラスで定義することをメソッドの (v) と呼ぶ。
- あるクラス型変数 `obj` を使ってメソッドを呼び出すとき、オブジェクト指向の特徴の 1 つである (w) により、`obj` の型のクラスではなく、それが (x) しているオブジェクトのクラスで定義されるメソッドを呼び出す。
- 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、バカソート (俗称)、(y)、バブルソート、クイックソート、マージソート、(z) などがあるが、データ数 N に対して、バブルソートの計算量のオーダーは (A) であり、ヒープソートの計算量のオーダーは (B) である。
- 可視性を決めるキーワードである `private` によって、宣言時に修飾されたフィールドやメソッドは、その (C) だけからしか参照できない。
- クラス `Student` のオブジェクトの参照情報 4 つを格納できる配列変数 `sL` の宣言、領域確保を行うためには、(D) と記述する。この時点で、`sL` のための `Student` のオブジェクトは生成されて (E) (いる or いない)。

問 2 . 電話やスマートフォンの入力デバイスの情報を扱う次のプログラムを作成した。このプログラムが表示するものを示せ。

```
class Kimatsu2 {
    public static void main(String[] args) {
        InputDevice idv1 = new InputDevice("西ブレ", true, false), idv2;
        TouchPanel tchp = new TouchPanel("三葉電子", 10);
        idv2 = tchp; idv1.printInf(); idv2.printInf(); tchp.printInf();
        Phone phn1 = new Phone("ZGMF-X10A", idv1);
        Phone phn2 = new Phone("RX78-II", idv2);
        SmartPhone phn3 = new SmartPhone("GN-0000", tchp);
        phn1.printInf(); phn2.printInf(); phn3.printInf();
    }
}
```

```

class Phone {
    String name;
    InputDevice inputDevice;
    Phone(String name, InputDevice inputDevice) {
        this.name          = name;
        this.inputDevice    = inputDevice;
    }
    void printInf() {
        System.out.println("この電話の名前は" + name + "である。");
        inputDevice.printInf();
    }
}

class SmartPhone extends Phone {
    SmartPhone(String name, TouchPanel tPanel) {
        super(name, tPanel);
    }
    void printInf() {
        System.out.println("このスマートフォンの名前は" + name + "である。");
        inputDevice.printInf();
    }
}

class InputDevice {
    String manufacture;
    boolean isCharacter;
    boolean isPosition;
    InputDevice(String manufacture, boolean isC, boolean isP) {
        this.manufacture = manufacture;
        isCharacter = isC;
        isPosition = isP;
    }
    void printInf() {
        if (isCharacter) System.out.println(manufacture + "製の文字入力デバイス");
        if (isPosition) System.out.println(manufacture + "製の座標入力デバイス");
    }
}

class TouchPanel extends InputDevice {
    int nFinger;
    TouchPanel(String manufacture, int nFinger) {
        super(manufacture, false, true);
        this.nFinger = nFinger;
    }
    void printInf() {
        super.printInf();
        System.out.println("タッチパネルです。");
    }
}

```

問3 . 次のプログラムは、3名のマスターと3名のサーバントがペアになって競う聖灰ゲームに関して、9組のペアを強いものから順番に表示するものである。その仕様を次に示す。

- (1) クラス Player は、マスターやサーバントの情報を記録する。クラス Player のフィールドは、名前を格納する String 型の変数 name と、攻撃力の値を格納する int 型の変数 aPoint, 防御力の値を格納する int 型の変数 dPoint, 幸運度の値を格納する int 型の変数 lPoint である。
- (2) Player のメソッドはコンストラクタ Player(String name, int aPoint, int dPoint, int lPoint) だけであり、それぞれの引数をフィールドに格納する。
- (3) クラス Pair はペアに関する情報の格納と計算を行う。クラス Pair のフィールドは、それぞれマスターとサーバントを表す Player 型の master と servant と、攻守力を表す int 型の tPoint である。
- (4) コンストラクタ Pair(Player master, Player servant) は、Player 型の引数をそれぞれのフィールドに格納し、ペアの攻守力を計算するメソッド呼び出し、その結果をフィールドの tPoint に格納する。
- (5) メソッド calTPoint() は、そのペアの攻守力を計算し、int 型の戻り値として返すメソッドである。攻守力は、マスターの防御力とサーバントの攻撃力を加算したものの2乗に、マスターの攻撃力とサーバントの防御力を加算したものである。
- (6) メソッド stronger(Pair pair1, Pair pair2) は、引数の pair1 が pair2 よりも強いときには true を返し、そうでないときは false を返す boolean を返り値とするクラスメソッドである。あるペアが他のペアより強いとは、攻守力の値が大きい場合、また攻守力の値が同じ場合にはマスターの幸運度が大きい場合とする。
- (7) メソッド print() は、ペアのマスターとサーバントの名前、攻守力、マスターの個運度を表示する。
- (8) クラス Kimatsu3 は、メインメソッドのためのクラスである。メインメソッド main(String[] args) で、3人のマスターと3人のマスターのための Player 型のオブジェクトの参照情報を格納する配列 masterL と servantL の宣言と領域確保をそれぞれ行い、与えられたデータを格納する。次に、すべてのペアのための Pairr 型のオブジェクトの参照情報を格納する配列 pairL の宣言と領域確保をそれぞれ行い、ペアの情報を格納する。そしてペアの配列を、選択ソートにより、強い順番に並び替える。強さが同じ場合の順番は任意で良い。最後に、ソートされた順番にすべてのペアの情報を表示する。
- (9) 3人のマスターと3人のサーバントの情報は以下のとおりである。

マスター				サーバント			
氏名	攻撃力	防御力	幸運度	氏名	攻撃力	防御力	幸運度
加藤 恵	3	2	2	衛宮 士郎	2	4	3
小鳥遊六花	2	3	3	高尾 和成	2	2	2
常守 朱	3	2	1	七瀬 遙	1	3	2

このとき、無駄なくできるだけ一般性を持たせて、上記の仕様を満たすために、プログラム中の下線部 (a) ~ (l) に書き入れるべきものを答えよ。

```
class Player {
    String name;
    int aPoint, dPoint, lPoint;
    Player(String name, int aPoint, int dPoint, int lPoint) {
        _____ (a) _____ name = new String(name);
        _____ (a) _____ aPoint = aPoint;
        _____ (a) _____ dPoint = dPoint;
        _____ (a) _____ lPoint = lPoint;
    }
}
```

(次ページに続く)

```

class Pair {
    Player master, servant;
    int tPoint;
    Pair(Player master, Player servant) {
        this.master = master;
        this.servant = servant;
        tPoint = _____ (b)
    }
    int calTPoint() {
        _____ (c)
    }
    static boolean stronger(Pair pair1, Pair pair2) {
        return pair1.tPoint > pair2.tPoint || (pair1.tPoint == pair2.tPoint
            && pair1.master.lPoint > pair2.master.lPoint);
    }
    void print() {
        System.out.printf("%5s - %5s %5d %5d \n", master.name, servant.name, tPoint, master.lPoint);
    }
}

class Kimatsu3 {
    public static void main(String[] args) {
        int nMaster = 3, nServant = 3;
        int nPair = nMaster * nServant;
        Player masterL[] = new Player[nMaster];
        Player servantL[] = new Player[nServant];
        masterL[0] = new Player("加藤  恵", 3, 2, 2);
        masterL[1] = new Player("小鳥遊六花", 2, 3, 3);
        masterL[2] = new Player("常守  朱", 3, 2, 1);
        servantL[0] = new Player("衛宮  士郎", 2, 4, 3);
        servantL[1] = new Player("高尾  和成", 2, 2, 2);
        servantL[2] = new Player("七瀬  遙", 1, 3, 2);
        Pair pairL[] = new Pair[nPair];
        /* すべてのペアを生成 */
        int l = 0;
        for (int i = 0 ; i < nMaster ; ++i) {
            for (int j = 0 ; j < nServant ; ++j) {
                pairL[_____ (d) _____] = new Pair(masterL[i], servantL[j]);
            }
        }
        /* 選択ソート */
        for (l = 0 ; l < _____ (e) _____ ; ++l) {
            int k = _____ (f) _____
            for (int m = _____ (g) _____ ; m < nPair ; ++m) {
                if (! Pair.stronger( _____ (h) _____ , pairL[m])) _____ (i) _____
            }
            Pair tmp = _____ (j) _____
            pairL[k] = _____ (k) _____
            pairL[l] = _____ (l) _____
        }
        for (int i = 0 ; i < nPair ; ++i) pairL[i].print();
    }
}

```

学科・類：

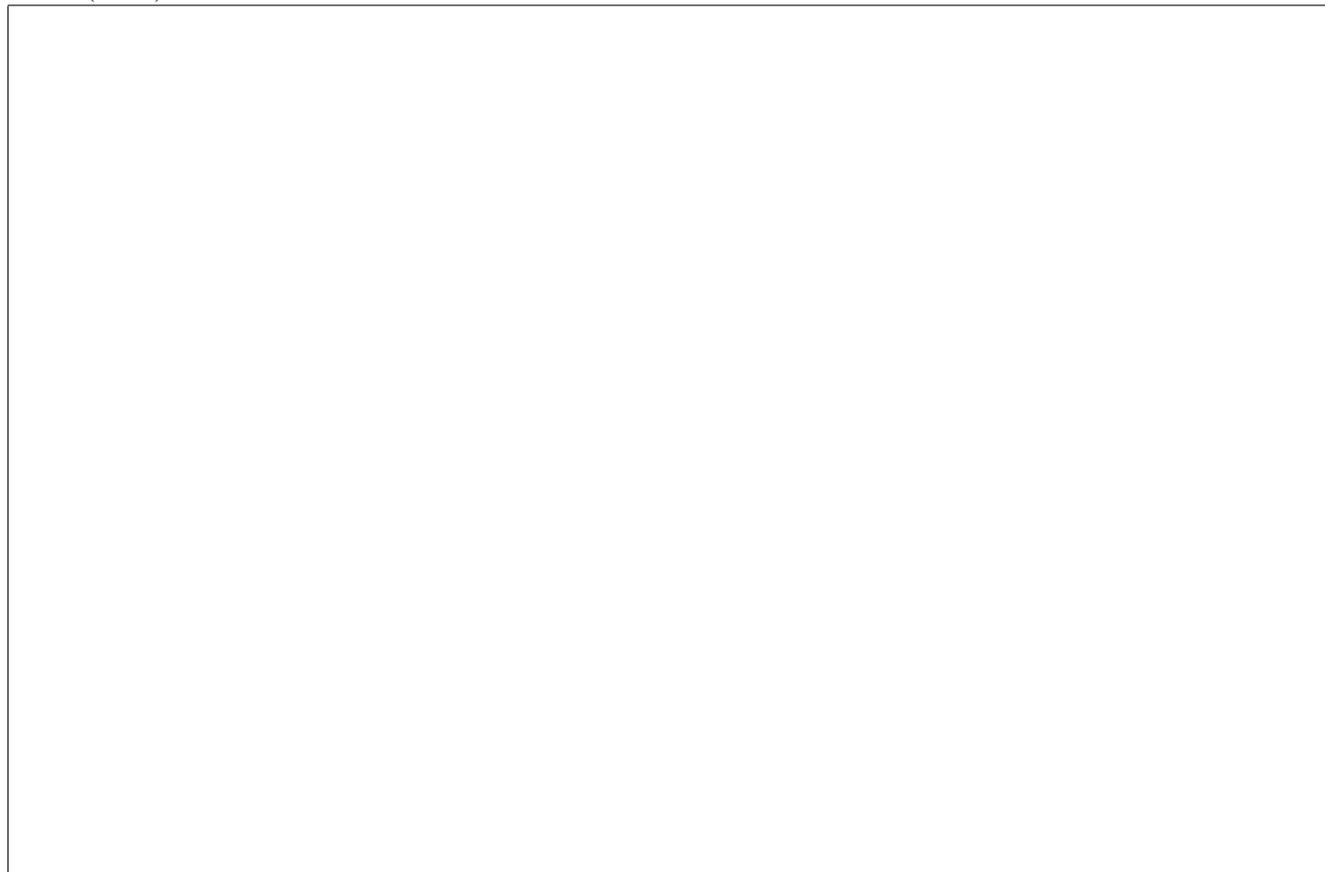
学籍番号：

名前：

問1 . $2 \times 31 = 62$ 点

- (a)
- (b)
- (c)
- (d)
- (e)
- (f)
- (g)
- (h)
- (i)
- (j)
- (k)
- (l)
- (m)
- (n)
- (o)
- (p)
- (q)
- (r)
- (s)
- (t)
- (u)
- (v)
- (w)
- (x)
- (y)
- (z)
- (A)
- (B)
- (C)
- (D)
- (E)

問2 . (14 点)



問3 . $2 \times 12 = 24$ 点

- (a)
- (b)
- (c)

- (d)
- (e)
- (f)
- (g)
- (h)
- (i)
- (j)
- (k)
- (l)

学科・類： 学籍番号： 名前： _____

問 1 . 2 × 31 = 62 点

- (a) `short iList[] = new short[5];`
- (b) `0 (iList[0])`
- (c) `4 (iList[4])`
- (d) `short iList[] = {3, 2, 3, 4, 8};`
- (e) シグネチャ
- (f) 名前 (識別子)
- (g) 引数
- (h) 型
- (i) オーバーロード
- (j) その変数を使うことができるプログラムにおける範囲
- (k) `this.v`
- (l) コンストラクタ
- (m) `Student std1 = new Student();`
- (n) `std1.pPhy = 123;`
- (o) `this`
- (p) 継承 (inheritance)
- (q) `Student`
- (r) スーパー
- (s) `UnivStudent`
- (t) サブ
- (u) `super`
- (v) オーバーライド
- (w) ポリモフィズム
- (x) 参照 (格納)
- (y) 選択ソート ((y) と (z) は順不同)
- (z) ヒープソート
- (A) N^2
- (B) $N \log_2 N$
- (C) 同じクラスのオブジェクト
- (D) `Student sL[] = new Student[4];`
- (E) いない。

問2 . (14 点)

西ブレ製の文字入力デバイス
三葉電子製の座標入力デバイス
タッチパネルです。
三葉電子製の座標入力デバイス
タッチパネルです。
この電話の名前は ZGMF-X10A である。
西ブレ製の文字入力デバイス
この電話の名前は RX78-II である。
三葉電子製の座標入力デバイス
タッチパネルです。
このスマートフォンの名前は GN-0000 である。
三葉電子製の座標入力デバイス
タッチパネルです。

問3 . $2 \times 12 = 24$ 点

- (a) this.
- (b) calTPoint();
- (c) return (master.dPoint + servant.aPoint) * (master.dPoint + servant.aPoint)
+ (master.aPoint + servant.dPoint);
- (d) l++
- (e) nPair - 1
- (f) l;
- (g) l + 1
- (h) pairL[k]
- (i) k = m;
- (j) pairL[k];
- (k) pairL[l];
- (l) tmp;