問 11　次の Java プログラムの説明及びプログラムを読んで，設問 1，2 に答えよ。
（Java プログラムで使用する API の説明は，この冊子の末尾を参照してください。）

〔プログラムの説明〕

試験の成績を管理するプログラムである。

(1) クラス ScoreManager は，後述するクラス ValueSortedMap を継承したクラスで，試験の成績を管理する。点数の高い順に学籍番号（英数字で構成される固定長の文字列）と点数を出力するメソッド print をもつ。

(2) クラス ValueSortedMap は，キーと値の対応付けを値の昇順，又はコンストラクタで指定したコンパレータに従った順に保持する。キーと値の対応付けをキーの昇順，又はコンパレータに従った順に保持するクラス TreeMap を利用している。主なメソッドは次のとおりである。

```
public V put(K key, V value)
```

key に value を対応付けて登録する。key 又は value が null のときは例外 NullPointerException を投げる。key が既に他の値と対応付けられていれば，その値を value で置き換え，置き換えられる前の値を返す。key に値が対応付けられていなければ，null を返す。

```
public V get(K key)
```

key に対応付けられた値を返す。key と値の対応付けがなければ，null を返す。

```
public V remove(K key)
```

key と値の対応付けを削除し，対応付けられていた値を返す。key と値の対応付けがなければ，null を返す。

```
public Iterator<K> iterator()
```

対応付けられた値の昇順，又はコンストラクタで指定したコンパレータに従った順に，キーを返すための反復子を返す。

フィールド map は，キーと値の対応付けを保持する。

フィールド reverseMap は，値にキーを対応付けて保持する。異なるキーに同じ値が対応付けられることがあるので，値に対応付けられるのはキーのリストである。

(3) クラス ScoreManagerTester は，成績管理プログラムのテストプログラムである。メソッド main の実行結果を，図 1 に示す。

```
数学の成績
 FE0002 : 90 点
 FE0005 : 90 点
 FE0003 : 85 点
 FE0001 : 85 点
null
```

図 1　メソッド main の実行結果

〔プログラム 1〕

```
import java.util.Comparator;

public class ScoreManager extends ValueSortedMap<  a  > {
   private final String subject;

   public ScoreManager(String subject) {
      super(new Comparator<Integer>() {
            public int compare(Integer a, Integer b) {
               return b.compareTo(a);
            }
         });
      this.subject = subject;
   }

   public void print() {
      System.out.println(subject + "の成績");
      for (String name : this) {
         System.out.printf(" %s : %d点%n", name, get(name));
      }
   }
}
```

〔プログラム 2〕

```
import java.util.ArrayList;
import java.util.Comparator;
import java.util.HashMap;
import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.NoSuchElementException;
import java.util.TreeMap;
```

```
public class ValueSortedMap<K, V> implements Iterable<K> {
   Map<K, V> map = new HashMap<K, V>();
   Map<V, List<K>> reverseMap;

   public ValueSortedMap() {
      reverseMap = new TreeMap<V, List<K>>();
   }

   public ValueSortedMap(Comparator<? super V> c) {
      reverseMap = new TreeMap<V, List<K>>(c);
   }

   public V put(K key, V value) {
      if ( [ b ] ) {
         throw new NullPointerException();
      }
      V old = remove(key);
      map.put(key, value);
      List<K> keys = reverseMap.get(value);
      if ( [ c ] ) {
         keys = new ArrayList<K>();
         reverseMap.put(value, keys);
      }
      keys.add(key);
      return old;
   }

   public V get(K key) {
      return map.get(key);
   }

   public V remove(K key) {
      V value = [ d ] (key);
      if (value != null) {
         List<K> keys = reverseMap.get(value);
         keys.remove(key);
         if (keys.isEmpty()) {
            [ e ] (value);
         }
      }
      return value;
   }

   public int size() {
      return map.size();
   }

   public Iterator<K> iterator() {
      return new Iterator<K>() {
         Iterator<V> vi = reverseMap.keySet().iterator();
```

```
        Iterator<K> ki = new ArrayList<K>().iterator();

        public boolean hasNext() {
            return vi.hasNext() || ki.hasNext();
        }

        public K next() {
            if (hasNext()) {
                if (!ki.hasNext()) {
                    ki = reverseMap.get(vi.next()).iterator();
                }
                return ki.next();
            }
            throw new NoSuchElementException();
        }

        public void remove() {
            throw new UnsupportedOperationException();
        }
    };
  }
}
```

〔プログラム3〕

```
public class ScoreManagerTester {
   public static void main(String[] args) {
      ScoreManager sm = new ScoreManager("数学");
      try {
         sm.put("FE0001", 70);
         sm.put("FE0002", 90);
         sm.put("FE0003", 85);
         sm.put("FE0004", 95);
         sm.put("FE0005", 90);
         sm.put("FE0001", 85);
         sm.remove("FE0004");
         sm.put(null, 90);
         sm.put("FE0004", 90);
      } catch (NullPointerException e) {}
      sm.print();
      System.out.println(sm.get("FE0004"));
   }
}
```

設問1　プログラム中の［　　］に入れる正しい答えを，解答群の中から選べ。

a に関する解答群

ア　Integer　　イ　Integer, String

ウ　String　　エ　String, Integer

b に関する解答群

ア　key != null && value != null　　イ　key != null || value != null

ウ　key == null && value == null　　エ　key == null || value == null

c に関する解答群

ア　!keys.isEmpty()　　イ　keys != null

ウ　keys == null　　エ　keys.isEmpty()

d, e に関する解答群

ア　Map.get　　イ　map.get

ウ　Map.remove　　エ　map.remove

オ　remove　　カ　reverseMap.get

キ　reverseMap.remove

設問2　点数が同じ場合には，学籍番号の文字列としての自然順序付けに従って出力するように変更する。クラス ValueSortedMap で使用しているクラスやインタフェースの変更だけで実現する場合，変更内容として適切なものを，解答群の中から選べ。

解答群

ア　ArrayList を TreeSet に変更する。

イ　List と ArrayList を Set に変更する。

ウ　List を Set に変更し，ArrayList を TreeSet に変更する。

エ　List を TreeSet に変更し，ArrayList を Set に変更する。

オ　List を TreeSet に変更する。