Xi'an Jiaotong-Liverpool University

西交利物浦大學

Paper CODE	EXAMINER	EMAIL OF EXAMINER	TEL
CPT105/2022/23 S1 Final Exam	Yushan Pan	Yushan.Pan@xjtlu.edu.cn	+86-(0)512-89165347

1st SEMESTER 2022/23 FINAL EXAMINATION

Undergraduate – Year 1

INTRODUCTION TO PROGRAMMING IN JAVA

TIME ALLOWED: 2 HOURS

INSTRUCTIONS TO CANDIDATES

1. **This is a CLOSED-book examination. Please tick the integrity disclaimer when handing in your answer sheets. Please complete the assessment independently and honestly.**
2. **The total marks are 100.**
3. **The exam paper consists of 19 printed pages.**
4. **This exam consists of 3 sections:**

   Section A consists of single-choice questions with 23 questions worth for a total mark of 25.

   Section B consists of multiple-choice questions with 10 questions worth three marks each for a total mark of 30.

   Section C consists of coding tasks with three questions. The first question is worth 10 marks. The second question is worth 20 marks. And the last one is worth 15 marks. In total, 45 marks for Section C.

   Answer all questions. There is NO penalty for providing a wrong answer.
5. **Only English solutions are accepted. Answers need to be handwritten on answer sheets. Please clearly indicate the question numbers before your solutions. Please also indicate ONLY your student ID number on answer sheets.**
6. **The exam duration is 2 hours. If any major problems are preventing you from continuing the exam or submitting your answers in time, please do not hesitate to email the Module Examiner or Assessment Team of Registry (assessment@xjtlu.edu.cn).**

Section A Single Choice (25 marks)

This section consists of 23 questions. From question 1 to 21, each question is worth 1 mark. Question 22 and 23 are worth 2 marks per each.

Note: There is only one correct answer for the following Single Choice questions. Red colour is the correct answer.

1. Suggest a = 2, b = 3. Which Boolean statement's value is True? (1 mark)

A. a >= b && a >= 1

B. (a >= 0 && b <= 0) || (a < 0 && b < 0)

C. (a < 0 && b < 0) || a <= b

D. a >= 0 && b >= 0 && a >= b

2. What is the output of the following program fragment? (1 mark)

```java
int a = 2;
if (a >= 1)
    System.out.print("Statement 1 ");
else if (a >= 2)
    System.out.print("Statement 2 ");
else if (a >= 3)
    System.out.print("Statement 3 ");
```

A. Statement 1

B. Statement 2

C. Statement 3

D. Statement 1 Statement 2

3. What is the output of the following program fragment? (1 mark)

```java
int count = 0;
for(int i = 0; i <= 10; i = i + 2)
    count += i;
System.out.print(count);
```

A. 55

B. 30

C. 25

D. 18

4. What is the output of the following program fragment? (1 mark)

```java
int count = 0, loop_int = 1;
while (loop_int <= 5){
    loop_int += 1;
    count += loop_int * (loop_int - 1);
}
System.out.print(count);
```

A. 40

B. 70

C. 112

D. 168

5. What is the output of the following program fragment? (1 mark)

```java
int count = 0;
for (int i = 1; i <= 5; i++)
    for (int j = 1; j <= i; j++)
        count += j * j;
System.out.print(count);
```

A. 105

B. 275

C. 55

D. 225

6. What is the output of the following program fragment? (1 mark)

```java
int[] a = {1, 1, 2, 3, 5, 8, 13, 21};
for (int n = 0; n < 5; n++) {
    a[n] = a[n+1];
    a[n+1] = a[n+2];
    a[n+2] = a[n+3];
    a[n+3] = a[n] + a[n+1] + a[n+2];
}
System.out.println(a[7]);
```

A. 42

B. 44

C. 78

D. 82

7. What is the possible output of the following program fragment? (1 mark)

```
int N = 10000000;
int count = 0;
double x, y;
for (int i = 0; i < N; i++) {
        x = Math.random( );
        y = Math.random( );
        if (x > y) count++;
}
System.out.println( count*1.0 / N );
```

A. 0

B. 1

C. 0.5

D. - 0.5

8. What is the output of the following program fragment? (1 mark)

```
String str = "1#";
int[] list = {1, 2, 3};
try{
    System.out.print(Integer.parseInt(str));
    System.out.print(list[3]);
}
catch (ArrayIndexOutOfBoundsException e){
    System.out.print("#Exception 1;");
}
catch (NumberFormatException e){
    System.out.print("#Exception 2;");
}
```

A. 13

B. #Exception 2;#Exception 1;

C. 1#Exception 1;

D. #Exception 2;

9. What is the output of the following program fragment? (1 mark)

```java
public class MyData {
    public static int i;
    public int j;

    public MyData(int input_a, int input_b){
        i = input_a;
        j = input_b;
    }

    public int returnSum(){
        return i + j;
    }

    public static void main(String[] args) {
        MyData myData1 = new MyData(1, 2);
        MyData myData2 = new MyData(2, 3);
        System.out.print(myData1.returnSum() + myData2.returnSum());
    }
}
```

A. 7

B. 8

C. 9

D. 10

10. What is the output of the following program fragment? (1 mark)

```java
public class FList {
    public static int getValue(int index){
        if (index == 0 || index == 1)
            return 1;
        else if (index == 2)
            return 2;
        else
            return getValue(index - 1) + getValue(index - 2);
    }

    public static void main(String[] args) {
        System.out.println(getValue(6));
    }
}
```

A. 5
B. 8
C. 13
D. 21

11. What is the output of the following program fragment? (1 mark)

```java
public class getNum {
    public static int getAimNum(int[] nums) {
        int a = -1000, b = -1000, c = -1000;
        for (int num : nums) {
            if (num > a) {
                c = b;
                b = a;
                a = num;
            } else if (a > num && num > b) {
                c = b;
                b = num;
            } else if (b > num && num > c) {
                c = num;
            }
        }
        return c == -1000 ? (int) a : (int) c;
    }

    public static void main(String[] args) {
        System.out.print(getAimNum(new int[]{3, 2, 1}));
        System.out.print(getAimNum(new int[]{2, 1}));
        System.out.print(getAimNum(new int[]{2, 3, 2, 1}));
    }
}
```

A. 121
B. 323
C. 322
D. 111

12. What is the output of the following program fragment? (1 mark)

```java
ArrayList<String> sites = new ArrayList<String>();
int count = 0;
sites.add("I");
sites.add("hope");
sites.add("do");
sites.add("well");
sites.add("in");
sites.add("exam");
for (String word : sites)
    for (int i = 0 ; i < word.length(); i++)
        if (word.charAt(i) == 'e')
            count ++;
System.out.print(count);
```

A. 3
B. 4
C. 5
D. 6

13. What is the output of the following program fragment? (1 mark)

```java
int count = 0;
for (int i = 0; i < 10; i++)
    for (int j = 0; j < i; j++)
        count += 1;
System.out.println(count);
```

A. 45
B. 35
C. 55
D. 65

14. What is the output of the following program fragment? (1 mark)

```java
int[][] matrix = {
        {1, 2, 3, 6, 2},
        {2, 3, 5, 7, 2},
        {9, 3, 2, 6, 2},
        {1, 6, 5, 7, 2},
        {1, 9, 2, 3, 6}
};
int max = -1;
for (int i = 0; i < 5; i++) {
    int temp = 0;
    for (int j = 0; j < 5; j++)
        temp += matrix[j][i];
    max = max > temp ? max : temp;
}
System.out.println(max);
```

A. 29

B. 22

C. 14

D. 20

15. What is the output of the following program fragment? (1 mark)

```java
public class Main {
    public static int getSum(int input) {
        if (input < 2)
            return -1;
        if (input == 2)
            return 0;
        if (input == 3)
            return 1;
        int c2 = getSum(input-2);
        int c3 = getSum(input-3);
        return c3 != -1 ? c3 + 1 : c2;
    }

    public static void main(String[] args) {
        System.out.println(getSum(17));
    }
}
```

A. 1

B. 5

C. -1

D. 3

16. What is the output of the following program fragment? (1 mark)

```
public class Main {
    public static int getSum(int input) {
        if (input < 2)
            return 0;
        if (input == 2 || input == 3)
            return 1;
        return getSum(input-2) + getSum(input-3);
    }
    public static void main(String[] args) {
        System.out.println(getSum(11));
    }
}
```

A. 6

B. 7

C. 8

D. 9

17. What is the output of the following program fragment? (1 mark)

```
String s = "12X_367_22A_DE2";
int count = 0, sum = 0;
for(int i = 0; i < s.length(); i++) {
    if (Character.isDigit(s.charAt(i))) {
        count ++;
        sum += Integer.parseInt(Character.toString(s.charAt(i)));
    }
}
System.out.println(sum % count);
```

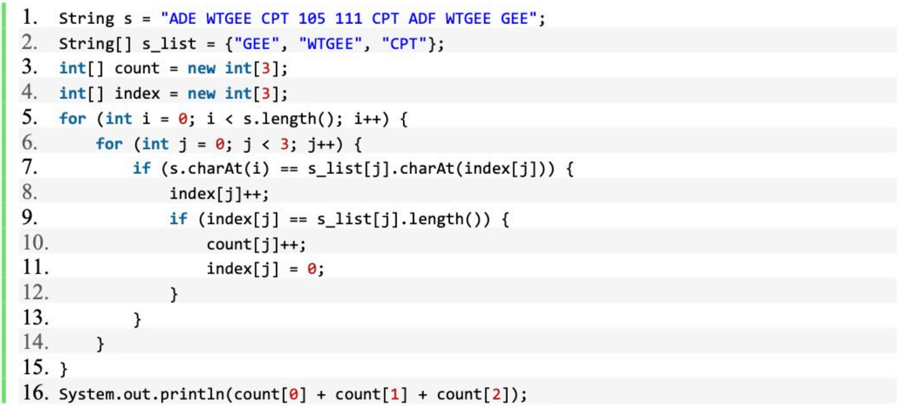
A. 0

B. 1

C. 2

D. 3

18. What is the output of the following program fragment? (1 mark)



```java
String s = "ADE WTGEE CPT 105 111 CPT ADF WTGEE GEE";
String[] s_list = {"GEE", "WTGEE", "CPT"};
int[] count = new int[3];
int[] index = new int[3];
for (int i = 0; i < s.length(); i++) {
    for (int j = 0; j < 3; j++) {
        if (s.charAt(i) == s_list[j].charAt(index[j])) {
            index[j]++;
            if (index[j] == s_list[j].length()) {
                count[j]++;
                index[j] = 0;
            }
        }
    }
}
System.out.println(count[0] + count[1] + count[2]);
```

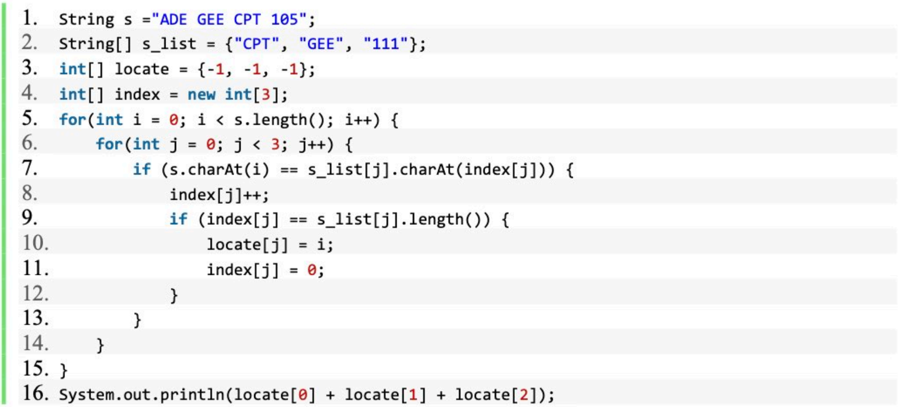
A. 0

B. 3

C. 5

D. 7

19. What is the output of the following program fragment? (1 mark)



```java
String s ="ADE GEE CPT 105";
String[] s_list = {"CPT", "GEE", "111"};
int[] locate = {-1, -1, -1};
int[] index = new int[3];
for(int i = 0; i < s.length(); i++) {
    for(int j = 0; j < 3; j++) {
        if (s.charAt(i) == s_list[j].charAt(index[j])) {
            index[j]++;
            if (index[j] == s_list[j].length()) {
                locate[j] = i;
                index[j] = 0;
            }
        }
    }
}
System.out.println(locate[0] + locate[1] + locate[2]);
```

A. 9

B. 14

C. 15

D. 16

20. What is the output of the following program fragment? (1 mark)

```java
public class CheckAllStr {
    public static int deleteCount(String[] strs) {
        int row = strs.length;
        int col = strs[0].length();
        int ans = 0;
        for (int j = 0; j < col; ++j) {
            for (int i = 1; i < row; ++i) {
                if (strs[i - 1].charAt(j) > strs[i].charAt(j)) {
                    ans++;
                    break;
                }
            }
        }
        return ans;
    }

    public static void main(String[] args) {
        String[] strList = {"abc", "cef", "abc"};
        System.out.println(deleteCount(strList));
    }
}
```

A. 0

B. 1

C. 2

D. 3

21. According to the code fragment below, which option’s return value is -1? (1 mark)

```java
public static int checkStr(String str1, String str2) {
    int n = str1.length(), m = str2.length();
    for (int i = 0; i + m <= n; i++) {
        boolean flag = true;
        for (int j = 0; j < m; j++) {
            if (str1.charAt(i + j) != str2.charAt(j)) {
                flag = false;
                break;
            }
        }
        if (flag) return i;
    }
    return -1;
}
```

A. checkStr("abcdef", "bcd")

B. checkStr("abcdef", "acd")

C. checkStr("abcdef", "cde")

D. checkStr("abcdef", "abcd")

Background for questions 22 and 23. Monster Game. We are writing the monster part's scripts of a game. According to the game design, there are 3 types of monsters in this game: land monsters, flying monsters, and deep-sea monsters. Each type of monster has a different behavioral pattern. Land monsters can run, dig holes and sleep. Flying monsters can fly, scout and be invisible. Deep sea monsters can swim, sleep, and escape. Monsters have three attack kinds: single attack, ranged attack, and summoning other monsters. Each monster may have between 0 and 3 types of attacks.

22. According to the above Monster Game's design. Which option below is the most suitable way for implement this game? (2 marks)

A. Create an Interface class for each monster kind. Create an Abstract class for each attack method.

B. Create an Interface class for each monster kind. Create an Interface class for each attack method.

C. Create an Abstract class for each monster kind. Create an Interface class for each attack method.

D. Create an Abstract class for each monster kind. Create an Abstract class for each attack method.

23. In anticipation of the upcoming Easter season, the game designers plan to add a new monster type to the Monster Game, the Easter Bunny. This monster has the following characteristics, can only move on land, can dig and sleep, cannot attack but has a very fast running speed. Now you are responsible for implementing this monster. Which of the following options is the best solution to implement? (2 marks)

A. Create a separate class for the Easter Bunny. This class implements all the properties of the Easter Bunny, including run, dig, sleep.

B. Consider Easter Bunny as a land monster, and use the existing land monster's class (interface or abstract) to implement it.

C. Consider Easter Bunny as a land monster, and implement the three skills, run, dig, and sleep.

D. Create a separate class for the Easter Bunny, and use the implement functions of run, dig, sleep in land monster class to achieve the Easter Bunny.

Section B Multiple Choice Questions (30 marks)

This section consists of 10 questions. Each question is worth three marks.

Note: There is more than one correct answer for each question. You can get a half score if your answer is incomplete; however, you will get no score if you choose a wrong answer.

1. According to the code fragment below, which option's return value is 6?

```java
public static int getAns(int x) {
    int l = 0, r = x, ans = -1;
    while (l <= r) {
        int mid = l + (r - l) / 2;
        if ((long) mid * mid <= x) {
            ans = mid;
            l = mid + 1;
        } else {
            r = mid - 1;
        }
    }
    return ans;
}
```

A. getAns(35)

B. getAns(36)

C. getAns(37)

D. getAns(38)

2. According to the code fragment below, which option's return value is true?

```java
public static boolean checkAllStr(String s) {
    int n = s.length();
    for (int i = 1; i <= n / 2; ++i) {
        if (n % i == 0) {
            boolean match = true;
            for (int j = i; j < n; ++j) {
                if (s.charAt(j) != s.charAt(j - i)) {
                    match = false;
                    break;
                }
            }
            if (match) return true;
        }
    }
    return false;
}
```

A. checkAllStr("abababababa")

B. checkAllStr("ababababab")

C. checkAllStr("abcdefg")

D. checkAllStr("abcdabcd")

3. According to the code fragment below, which option’s return value is true?

```java
public static boolean checkTwoStr(String str1, String str2) {
    int m = str1.length(), n = str2.length();
    if (m != n)  return false;
    for (int i = 0; i < n; i++) {
        boolean flag = true;
        for (int j = 0; j < n; j++) {
            if (str1.charAt((i + j) % n) != str2.charAt(j)) {
                flag = false;
                break;
            }
        }
        if (flag)  return true;
    }
    return false;
}
```

A. checkTwoStr("hello", "ellho")

B. checkTwoStr("hello", "elloh")

C. checkTwoStr("CPT105", "105CPT")

D. checkTwoStr("CPT105", "C1P0T5")

4. According to the code fragment below, which option has the correct output corresponding to the input?

```java
public static String changeStr(String str, char c) {
    int index = str.indexOf(c);
    if (index >= 0) {
        char[] arr = str.toCharArray();
        int left = 0, right = index;
        while (left < right) {
            char temp = arr[left];
            arr[left] = arr[right];
            arr[right] = temp;
            left++;
            right--;
        }
        str = new String(arr);
    }
    return str;
}
```

A. Input: changeStr("hello everyone", 'v') Output: e ollehveryone

B. Input: changeStr("never give up", 'i') Output: ig revenve up

C. Input: changeStr("believe yourself", 'e') Output: esruoy eveileblf

D. Input: changeStr("good luck", 'l') Output: l dooguck

5. According to the code fragment below, which mistakes may appear if the value of input is a string of at least 4 in length?

```java
public static int getSum(String input) {
    int count = 0;
    for(int i = 0; i < 5; i++)
        count += Integer.parseInt(Character.toString(input.charAt(i)));
    return count;
}
```

A. NumberFormatException

B. StringIndexOutOfBoundsException

C. NullPointerException

D. NegativeArraySizeException

6. According to the code fragment below, which option's return value is 105?

```java
public static int getSum(String input) {
    int count = 0;
    for(int i = 0; i < input.length(); i++) {
        if (Character.isDigit(input.charAt(i)) ||
                ( i + 1 < input.length() &&
                        input.charAt(i) == '-' &&
                        Character.isDigit(input.charAt(i + 1)))) {
            int len = 1;
            for (int j = i + 1; j < input.length(); j++) {
                if (Character.isDigit(input.charAt(j)))
                    len++;
                else
                    break;
            }
            count += Integer.parseInt(input.substring(i, i + len));
            i += len - 1;
        }
    }
    return count;
}
```

A. getSum(“105, --10 10”)

B. getSum(“25,33-10-3-20#80”)

C. getSum(“105-33-10-3-20”)

D. getSum(“105#33-10-3-20”)

7. According to the code fragment below, which option’s return value is 5?

```java
public static int getCount(String input) {
    if (input.length() == 1)
        return 0;
    return input.charAt(0) == input.charAt(1) ?
            getCount(input.substring(1)) :
            1 + getCount(input.substring(1));
}
```

A. getCount(“acbbb”)

B. getCount(“a#c#bbb#”)

C. getCount(“#111#12”)

D. getCount(“#11112”)

8. According to the code fragment below, which option’s return value is 5?

```java
public static int getArise(String input) {
    if (input.length() <= 1) return input.length();
    int count = 1, max;
    for (int i = 1; i < input.length(); i++)
        if (Integer.parseInt(Character.toString(input.charAt(i))) >=
                Integer.parseInt(Character.toString(input.charAt(i-1)))) count++;
        else break;
    max = Math.max(count, getArise(input.substring(1)));
    return Math.max(max, getArise(input.substring(0, input.length()-1)));
}
```

A. getArise(“12345”)

B. getArise(“1235545”)

C. getArise(“12312345”)

D. getArise(“54321”)

9. According to the code fragment below, which option’s return value is largest (maybe equal)?

```java
public static int getSumPlus(String input) {
    if (input.length() <= 1) return input.length();
    int max = -1;
    for (int i = 1; i < input.length(); i++)
        max = Math.max(max, Integer.parseInt(input.substring(0, i))+Integer.parseInt(input.substring(i)));
    return max;
}
```

A. getSumPlus(“123”)

B. getSumPlus(“231”)

C. getSumPlus(“312”)

D. getSumPlus(“112”)

10. According to the code fragment below, which option’s return value is 3?

```java
public static int getTwo(int input) {
    if (input == 0) return 0;
    if (input <= 2) return 1;
    int current = 1;
    while (true) {
        if (current <= input && current * 2 > input)
            return 1 + getTwo(input - current);
        current *= 2;
    }
}
```

A. getTwo(19)

B. getTwo(5)

C. getTwo(7)

D. getTwo(15)

Section C Coding (45 marks)

1. (10 marks) Filling blanks.

Complete the following code fragments where you can use nested for-loops to output the following pattern. You only need to write those blanks on your exam sheets. Blanks 1, 4, 5 and 7 are worth 1 mark and Blanks 2, 3, and 6 are worth 2 marks.

```
    #
   ##
  ###
 ####
#####
```

Code Fragment 1:

```
for(int i = 0; i BLANK1 5; i++){
    for(int j = 0; j < BLANK2 ; j++)
        System.out.print(" ");
    for(int j = 0; j BLANK3 i; j++)
        System.out.print("#");
    BLANK4
}
```

Code Fragment 2:

```
for(int i = 0; i < 5; i++){
    for(int j = 0; j BLANK5 5; j++)
        if(j < BLANK6)
            System.out.print(" ");
        else
            System.out.print("#");
    BLANK7
}
```

2. (20 marks) Leap Year Calculator

Write a method **isLeapYear** with a parameter of type int named **year**.

The parameter needs to be **greater than or equal to 1** and less than or equal to 9999.

If the parameter is not in that range return **false**.

Otherwise, if it is in the valid range, calculate if the year is a leap year and return **true** if it is a leap year, otherwise return **false**.

To determine whether a year is a leap year, follow these steps:
1. If the year is **evenly divisible by 4**, go to step 2. Otherwise, go to step 5.
2. If the year is **evenly divisible by 100**, go to step 3. Otherwise, go to step 4.
3. If the year is **evenly divisible by 400**, go to step 4. Otherwise, go to step 5.
4. The year is a leap year (it has 366 days). The method **isLeapYear** needs to return **true.**
5. The year is not a leap year (it has 365 days). The method **isLeapYear** needs to return **false**.

The following years **are not leap years**:
1700, 1800, 1900, 2100, 2200, 2300, 2500, 2600
This is because they are evenly divisible by 100 but not by 400.

The following years **are leap years**:
1600, 2000, 2400
This is because they are **evenly divisible by both 100 and 400**.

Examples of input/output:

- **isLeapYear(-1600);** → should **return false** since the parameter is **not in range (1-9999)**
- **isLeapYear(1600);** → should **return true** since 1600 is a leap year
- **isLeapYear(2017);** → should return false since 2017 is **not** a leap year
- **isLeapYear(2000);** → should **return true** because 2000 is a leap year

NOTE: The method **isLeapYear** needs to be defined as **public static** like we have been doing so far in the course.
NOTE: Do not add a **main** method to solution code.

3. (15 marks) Binary Search.

This section consists of 5 sub-questions.

Binary search is a query algorithm with extremely fast query speed. The following program fragment is a way to implement the binary search. Please answer the questions according to this program fragment.

```java
public static int f(int[] arr, int num, int start, int end){
    int mid = (end-start)/2 + start;
    return (arr[mid] == num || start > end) ?
            (start > end ? -1 : mid) :
            (num < arr[mid] ? f(arr,num,start,mid-1) : f(arr,num,mid+1,end));
}
```

Hint. Writing recursive procedures is generally effective in helping you understand and compute a recursive program. Take the following program fragment for the Fibonacci series (1, 1, 2, 3, 5, 8, ...) as an example. This fragment is a dichotomous lookup code which you can use for the following questions.

```java
public static int fi(int index) {
    return (index == 1 || index == 2) ? 1 :  f(index - 1) + f(index - 2);
}
```

If you want to calculate the value of f(5), then you can write like this:
fi(5) = fi(4) + fi(3) = fi(3) + fi(2) + fi(2) + fi(1) = fi(2) + fi(1) + 1 + 1 + 1 = 5

Question1 (3 marks). Based on the above dichotomous lookup code, please give the result of the following input. Please write the recursive procedures.
f(new int[]{1,3,5,7,9}, 3, 0, 4)

Question2 (3 marks). Based on the above dichotomous lookup code, please give the result of the following input. Please write the recursive procedures.
f(new int[]{1,3,5,7,9}, 6, 0, 4)

Question3 (2 marks). Based on the above dichotomous lookup code, please give the result of the following input. Please write the recursive procedures.
f(new int[]{1,3,5,9,9}, 6, 0, 4)

Question4 (4 marks). Based on the above dichotomous lookup code, please give the result of the following input. Please write the recursive procedures.
f(new int[]{1,3,9,5,7}, 5, 0, 4)

Question5 (3 marks). Write the necessary condition for the above binary search code fragment to work correctly.

--End of Paper--