NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

EXAMINATION FOR

Semester 1: 2006/7

CS1101X – PROGRAMMING METHODOLOGY

November 2006    Time allowed: 2 hours

INSTRUCTIONS TO CANDIDATES

1. This examination paper consists of TWENTY SIX (26) questions and comprises EIGHTEEN (18) printed pages.
2. This is an OPEN BOOK examination.
3. Answer all questions.
4. Write your answers in the ANSWER SHEETS provided.
5. Fill in your Matriculation Number clearly on every page of your ANSWER SHEETS.
6. You may use pencil to write your codes. Pen is preferred for other questions.
7. You must submit only the ANSWER SHEETS and no other document.
SECTION A (20 Multiple Choice Questions: 60 Marks)
Each question has one correct answer. Write your answer in the space provided on the ANSWER SHEETS. 3 marks for each correct answer and no penalty for wrong answer.

You are advised not to spend more than an hour on this section.

1. We want to assign grades to scores according to the following table:

<table>
<thead>
<tr>
<th>Score</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 80</td>
<td>“A”</td>
</tr>
<tr>
<td>≥ 70 but &lt; 80</td>
<td>“B”</td>
</tr>
<tr>
<td>≥ 60 but &lt; 70</td>
<td>“C”</td>
</tr>
<tr>
<td>≥ 50 but &lt; 60</td>
<td>“D”</td>
</tr>
<tr>
<td>&lt; 50</td>
<td>“F”</td>
</tr>
</tbody>
</table>

Which of the following code fragments correctly assigns a grade (a string) for a non-negative integer score, assuming that all variables have been declared?

i. ```
   if (score >= 80) grade = "A";
   if (score >= 70) grade = "B";
   if (score >= 60) grade = "C";
   if (score >= 50) grade = "D";
   if (score >= 0)  grade = "F";
```   

ii. ```
    if (score >= 0)  grade = "F";
    if (score >= 50) grade = "D";
    if (score >= 60) grade = "C";
    if (score >= 70) grade = "B";
    if (score >= 80) grade = "A";
```  

iii. ```
    if (score >= 80) grade = "A";
    else if (score >= 70) grade = "B";
    else if (score >= 60) grade = "C";
    else if (score >= 50) grade = "D";
    else if (score >= 0)  grade = "F";
```  

iv. ```
    if (score >= 0)  grade = "F";
    else if (score >= 50) grade = "D";
    else if (score >= 60) grade = "C";
    else if (score >= 70) grade = "B";
    else if (score >= 80) grade = "A";
```  

A. Only (i) and (ii)
B. Only (ii) and (iii)
C. Only (iii) and (iv)
D. Only (i) and (iii)
E. Only (ii) and (iv)
2. What is the output of the following code?

```java
int sum[][] = new int[4][4];
for (int k = 0; k < 4; k++) sum[k][0] = 1;
for (int k = 0; k < 4; k++) sum[0][k] = 1;
for (int m = 1; m < 4; m++)
    for (int n = 1; n < 4; n++)
        sum[m][n] = sum[m-1][n] + sum[m][n-1];
for (int n = 1; n < 4; n++)
    System.out.print(sum[3][n] + " ");
```

A. 4 4 4
B. 4 5 6
C. 4 6 10
D. 4 10 20
E. 5 15 35

3. What is the output of the following code?

```java
int a0=1, a1=1, a2=1, a3=1;
for (int n = 1; n < 8; n++) {
    a0 = a1;
    a1 = a2;
    a2 = a3;
    a3 = a0+a1+a2;
}
System.out.println(a3);
```

A. 57
B. 58
C. 105
D. 106
E. 107
4. Which of the following statements are true?

   i. Rows of a 2D array can be of different lengths.
   ii. Arrays can be created for primitive types as well as for objects.
   iii. A method can have array parameters.
   iv. A method’s return type can be an array.
   v. An existing array’s length can be extended by writing to the element one position beyond the existing last element.

A. Only (i), (ii) and (iii)
B. Only (ii), (iii) and (iv)
C. Only (i), (iii) and (iv)
D. Only (i), (ii), (iii) and (iv)
E. All of them.

5. A polynomial \( p(x) = ax^5 + bx^4 + cx^3 + dx^2 + ex + f \) has its coefficients \( a, b, c, d, e, \) and \( f \) stored in an array \( \text{poly} \) where \( \text{poly}[5]=a, \text{poly}[4]=b, \text{poly}[3]=c, \text{poly}[2]=d, \text{poly}[1]=e, \) and \( \text{poly}[0]=f. \)

Assuming that \( x \) and the coefficients are integers, which of the following codes evaluates \( p(x) \) for a given value of \( x \)?

A. \[
\text{int } i, \text{ ans } = 0; \\
\text{for } (i = \text{poly.length}; i > 0; i--)
\text{ ans } = (\text{ans } + \text{poly}[i]) \times x;
\]

B. \[
\text{int } i, \text{ ans } = 0; \\
\text{for } (i = \text{poly.length}; i > 0; i--)
\text{ ans } = (\text{ans } + \text{poly}[i - 1]) \times x;
\]

C. \[
\text{int } i, \text{ ans } = 0; \\
\text{for } (i = \text{poly.length}; i > 1; i--)
\text{ ans } = (\text{ans } + \text{poly}[i - 1]) \times x; \\
\text{ans } += \text{poly}[i];
\]

D. \[
\text{int } i, \text{ ans } = 0; \\
\text{for } (i = \text{poly.length}; i > 1; i--)
\text{ ans } = (\text{ans } + \text{poly}[i - 1]) \times x; \\
\text{ans } += \text{poly}[i - 1];
\]

E. None of the above
6. As in the way polynomial is stored in the previous question, we now have 2 polynomials \( p(x) \) and \( q(x) \) with their coefficients stored in arrays \( \text{polyA} \) and \( \text{polyB} \) respectively. Which of the following codes computes the product of the two polynomials?

Note that the product of \( p(x) = ax^5 + bx^4 + cx^3 + dx^2 + ex + f \) and \( q(x) = gx^5 + hx^4 + ix^3 + jx^2 + kx + l \) is polynomial \( r(x) = (ag)x^0 + (ah + bg)x^1 + (ai + bh + cg)x^2 + (aj + bi + ch + dg)x^3 + \ldots \)

\[
\begin{align*}
\text{i. } & \quad \text{int } r[] = \text{new int[ polyA.length + polyB.length - 1]; } \\
& \quad \text{for (int } i=r.length-1; i >= 0; i--) \\
& \phantom{xx} r[i] = 0; \\
& \quad \text{for (int } i=\text{polyA.length-1}; i >= 0; i--) \\
& \phantom{xx} \text{for (int } j=\text{polyB.length-1}; j >= 0; j--) \\
& \phantom{xxxx} r[i+j] += \text{polyA}[i] \times \text{polyB}[j];
\end{align*}
\[
\begin{align*}
\text{ii. } & \quad \text{int } r[] = \text{new int[ polyA.length + polyB.length - 1]; } \\
& \quad \text{for (int } i=r.length-1; i >= 0; i--) \\
& \phantom{xx} r[i] = 0; \\
& \quad \text{for (int } i=\text{polyA.length-1}; i >= 1; i--) \\
& \phantom{xx} \text{for (int } j=\text{polyB.length-1}; j >= 1; j--) \\
& \phantom{xxxx} r[i+j] += \text{polyA}[i] \times \text{polyB}[j];
\end{align*}
\[
\begin{align*}
\text{iii. } & \quad \text{int } r[] = \text{new int[ polyA.length + polyB.length - 1]; } \\
& \quad \text{for (int } i=0; i < r.length; i++) \\
& \phantom{xx} r[i] = 0; \\
& \quad \text{for (int } i=0; i < \text{polyA.length}; i++) \\
& \phantom{xx} \text{for (int } j=0; j < \text{polyB.length}; j++) \\
& \phantom{xxxx} r[i+j] += \text{polyA}[i] \times \text{polyB}[j];
\end{align*}
\[
\begin{align*}
\text{iv. } & \quad \text{int } r[] = \text{new int[ polyA.length + polyB.length - 1]; } \\
& \quad \text{for (int } i=0; i < r.length; i++) \\
& \phantom{xx} r[i] = 0; \\
& \quad \text{for (int } i=1; i < \text{polyA.length-1}; i++) \\
& \phantom{xx} \text{for (int } j=1; j < \text{polyB.length-1}; j++) \\
& \phantom{xxxx} r[i+j] += \text{polyA}[i] \times \text{polyB}[j];
\end{align*}
\]

A. Only (i) and (iii)  
B. Only (ii) and (iii)  
C. Only (i) and (iv)  
D. Only (ii) and (iv)  
E. None of the above combinations
7. What is the output of the following code with input: 3, 6, 9, 6, 12?

```java
int f[] = {2, 4, 6, 8, 10, 12, 14, 16, 18, 20};
int input, temp;
Scanner scanner = new Scanner(System.in);
for (int i=0; i < 5; i++) {
    input = scanner.nextInt();
    for (int j=0; j < f.length; j++){
        if (input == f[j]) {
            // once found, shift items around
            temp = f[j];
            for (int k=j; k > 0; k--)
                f[k] = f[k-1];
            f[0] = temp;
        }
    }
}
// output
for (int j=0; j < f.length; j++)
    System.out.print(f[j]+ " ");
```

A. 4 2 8 6 12 10 16 14 20 18
B. 12 6 2 4 8 10 14 16 18 20
C. 6 12 2 4 8 10 14 16 18 20
D. 2 4 6 8 10 12 14 16 18 20
E. 20 18 16 14 12 10 8 6 4 2
8. What is the output of the following code?

```java
public class A {
    public int i = 0;

    private A methodB(A a) {
        a = new A();
        a.i = 10;
        return a;
    }

    private A methodC(A a) {
        a.i = 20;
        return a;
    }

    public static void main(String[] args) {
        A c = new A();
        c.methodB(c);
        System.out.print(c.i + " ");
        c.methodC(c);
        System.out.println(c.i);
    }
}
```

A. 0 0  
B. 0 10  
C. 0 20  
D. 10 10  
E. 10 20  

9. Which of the following statements is true?

A. In Java, a class can be a subclass of two or more superclasses.  
B. All protected instance members of a class are visible to its superclass.  
C. Private instance members of a class are visible to all its subclasses.  
D. In an abstract class, only the abstract methods can be over-ridden and the non-abstract methods cannot be over-ridden.  
E. None of the above.
10. Given this regular expression

"x(xx)*((yy)*(zz)+)*[0-9]{1,3}"

How many of the following strings match the above regular expression?

"xxxxyy8"
"xxxxzzyzz99"
"xxxxyyzz5"
"xzyy34"
"xzz123"
"x0"

A. 2  
B. 3  
C. 4  
D. 5  
E. 6

11. What is the output of the following code?

```java
int[] arr = {1, 0, 0, 0, 0};
for (int i=0; i < arr.length - 1; i++)
    for (int j=i+1; j < arr.length; j++)
        arr[j] += arr[j-1];
for (int k=0; k < arr.length; k++)
    System.out.print(arr[k] + " ");
```

A. 1 2 3 4 5  
B. 1 1 2 5 9  
C. 1 1 2 5 14  
D. 1 2 5 14 28  
E. 1 2 5 14 42
12. What is the output of the following code?

```java
int[] g = {0, 0, 0, 0, 0, 0, 0, 0, 1, 1};
for (int i=1; i < g.length; i*=2) {
    for (int j=g.length-1; j >= i; j--) {
        g[j] += g[j - i];
    }
}
int tail=0;
for (int i=1; i < g.length; i++) {
    if (g[tail] != g[i]) {
        g[tail] = i;
        tail = i;
    }
}
for (int i=0; i < g.length; i++) {
    System.out.print(g[i] + " ");
}
```

A. 8 0 0 0 0 0 0 0 2 2
B. 8 0 0 0 0 0 0 0 8 2
C. 8 0 0 0 0 0 0 0 9 2
D. 8 0 0 0 0 0 0 0 8 8
E. 8 0 0 0 0 0 0 0 9 9

13. What is the output of the following code?

```java
String tmp, str = "abcdef";
int length = str.length();
for (int i=0; i < length - 1; i++) {
    tmp = str.substring(i, i+1);
    str = str.substring(1) + tmp;
}
System.out.println(str);
```

A. abcdef
B. eabcdf
C. fedcba
D. fabcde
E. faceae
14. Given the following code, what is mystery(987654)?

```java
public static int mystery (int n) {
    if (n == 0) return 0;
    else return n%10 + mystery(n/100);
}
```

A. 18  
B. 21  
C. 39  
D. 54  
E. 228

15. What is the output of the following program fragment?

```java
for (int i = 1; i <= 2; i++) {
    try {
        System.out.print( "A" );
        if ( i == 1 ) throw new Exception();
        System.out.print( "B" );
        if ( i == 2 ) throw new ArithmeticException();
        System.out.print( "C" );
    } catch ( ArithmeticException e ) {
        System.out.print( "D" );
        break;
    } catch ( Exception e ) {
        System.out.print( "E" );
    } finally {
        System.out.print( "F" );
    }
}
```

A. AEABD  
B. AEABDF  
C. AEFABD   
D. AEFABDF  
E. AEBCFABDCF
16. In the program fragment below, which of the outputs is the best estimated value?

```java
double sum = 0;
int N = 1000000;
for (int i = 0; i < N; i++)
    sum += 10 * Math.random() + 5 * Math.random() + 3;
System.out.println( sum / N );
```

A. 9.0  
B. 10.5  
C. 18.0  
D. 3.0  
E. There is no way to estimate.

17. This question is on sorting an unsorted integer array `x` of distinct elements while knowing the rank of each element of `x`. The smallest element in `x` has a rank of 1, the second smallest element has a rank of 2, and so on. The ranks of the elements are stored in another integer array `rank`. That is, we have `rank[i]` indicating the rank of `x[i]`. (For example, if `rank[0]` is 3, it means that `x[0]` is the third smallest element in array `x`.)

With inputs of `x` and `rank`, the method `rankSort` shown below is to output a new integer array `y` that contains the elements of `x` sorted in ascending order. Complete the method by choosing the correct statement to fill in the empty box.

```java
public static int[] rankSort( int[] x, int[] rank ) {
    int[] y = new int[ x.length ];
    for ( int i = 0; i < x.length; i++ )
        return y;
}
```

A. `y[ rank[i] - 1 ] = x[i];`
B. `y[ rank[i] - 1 ] = x[ rank[i] - 1 ];`
D. `y[i] = x[ rank[i] - 1 ];`
E. `y[i] = rank[ x[i] ] - 1;`
18. Given the following definitions of classes AA and BB. What is the output of the program?

```java
class AA {
    public AA() {
        this( 0 );
        System.out.print( "A1." );
    }

    public AA( int i ) {
        System.out.print( "A2." );
    }
}

class BB extends AA {
    public BB() {
        this( 0 );
        System.out.print( "B1." );
    }

    public BB( int i ) {
        System.out.print( "B2." );
    }

    public static void main( String[] args ) {
        BB b = new BB();
    }
}
```

19. Given the following definitions of classes AAA, BBB and CCC. What is the output of the Test program?

```java
class AAA {
    protected void f() {
        System.out.print( "A.f " );
    }

    public void g() {
        f();
        System.out.print( "A.g " );
    }
}

class BBB extends AAA {
    protected void f() {
        System.out.print( "B.f " );
    }
}

class CCC extends BBB {
    public void g() {
        f();
        System.out.print( "C.g " );
    }
}

class Test {
    public static void main( String[] args ) {
        AAA p = new AAA();
        AAA q = new BBB();
        BBB r = new CCC();

        p.g();
        q.g();
        r.g();
    }
}
```

A. A.f A.g A.f A.g B.f A.g
B. A.f A.g A.f A.g B.f C.g
C. A.f A.g B.f A.g B.f C.g
D. B.f C.g B.f C.g B.f C.g
E. The program will cause compile-time error.
20. The following is a method to sort an array of strings in lexicographic order (dictionary order). Some of the elements in the input array are allowed to be the null reference, and the method sorts the array such that all the null elements are moved to the end of the array. For example, if the input array is {null, "gh", null, "cd", "ab", "ij", null, "ef"}, after sorting, it becomes {"ab", "cd", "ef", "gh", "ij", null, null, null}. Complete the method by choosing the correct statements to fill in the two empty boxes (1) and (2).

```java
public static void strBubbleSort( String[] strs ) {
    int bottom = strs.length - 2;
    boolean exchanged;

    do {
        exchanged = false;
        for ( int i = 0; i <= bottom; i++ ) {
            boolean mustSwap = false;

            if ( strs[i] == null && strs[i+1] != null )
                (1)
            else if ( strs[i] == null || strs[i+1] == null )
                (2)
            else if ( strs[i].compareTo( strs[i+1] ) > 0 )
                mustSwap = true;

            if ( mustSwap ) {
                // Exchange.
                String temp = strs[i];
                strs[i] = strs[i+1];
                strs[i+1] = temp;
                exchanged = true;
            }
        }
        bottom--;
    } while ( exchanged );
}
```

A. (1) strs[i] = strs[i+1];  (2) mustSwap = false;
B. (1) mustSwap = false;  (2) mustSwap = false;
C. (1) mustSwap = false;  (2) mustSwap = true;
D. (1) mustSwap = true;  (2) mustSwap = true;
E. (1) mustSwap = true;  (2) mustSwap = false;
SECTION B (4 Questions: 8 Marks)

Write your answer in the space provided on the ANSWER SHEETS. 2 marks for each correct answer.

21. Given three distinct points \( A = (a_x, a_y), B = (b_x, b_y), \) and \( C = (c_x, c_y), \) what can be said about these three points if the determinant of this matrix
\[
\begin{vmatrix}
 a_x & a_y & 1 \\
 b_x & b_y & 1 \\
 c_x & c_y & 1
\end{vmatrix}
\]
is zero?

22. To decide whether there are any duplicate values in an array (of over 100 entries), we do not need to first sort the array. TRUE or FALSE?

23. To find the 5\textsuperscript{th} largest element in an array (of over 100 entries), we do not need to first sort the array. TRUE or FALSE?

24. To output both the largest and the smallest elements in an array (of over 100 entries), we must first sort the array. TRUE or FALSE?

SECTION C

25. Recursion [9 marks]

(a) Given the following recursive method, what values do the following compute?
\[ f(5, 1), f(128, 8), f(14, 5) \text{ and } f(5, 14). \] [4 marks]

```java
// Precondition: n >= 0
public static int f(int m, int n) {
    if (n==0) return m;
    else if (n%2 == 0)
        return f(m/2, n/2);
    else
        return f(m-1, n-1);
}
```

(b) Write a recursive code \texttt{stars(n)} to print \( n \) rows of asterisks (*), where the first row contains 1 asterisk, the second row 3 asterisks, the third row 5 asterisks, and so on. The following example is the output for \texttt{stars(5)}: [5 marks]

```
*
***
*****
*******
*********
```

(No mark will be awarded if you do not use recursion.)
26. *Arrays* [23 marks]

A **Card** class is defined below. A card can be of one of these four suits: Club ♣, Diamond ♦, Heart ♥, or Spade ♠, and one of these 13 face values (or simply faces): Ace, 2…9, 10, Jack, Queen, and King. In the **Card** class, the suit is represented by a String, and a face by a character (‘A’, ‘2’ … ‘9’, ‘T’, ‘J’, ‘Q’, and ‘K’).

```java
// Card.java: This class defines a playing card.

class Card {

    // Data members
    private String suit; // "Club", "Diamond", "Heart", "Spade"
    private char   face; // 'A', '2'...'9', 'T', 'J', 'Q', 'K'

    // Constructors
    public Card() {
        this("unknown", ' ');
    }

    public Card(Card c) {
        this(c.getSuit(), c.getFace());
    }

    public Card(String s, char f) {
        setSuit(s);
        setFace(f);
    }

    // Get suit of this card
    public String getSuit() {
        return this.suit;
    }

    // Get face of this card
    public char getFace() {
        return this.face;
    }

    // Set suit of this card
    public void setSuit(String s) {
        this.suit = s;
    }

    // Set face of this card
    public void setFace(char f) {
        this.face = f;
    }

    // Returns string representation of a Card object
    public String toString() {
        return "[" + this.getSuit() + "-" + this.getFace() + "]";
    }
}
```
26. (cont...) A partial code **CardGame.java** is given below.

```java
// CardGame.java
import java.util.*;

class CardGame {
    static private Card[] cards = new Card[52];
    private Card[][] hands;

    public CardGame () {
        this.hands = new Card[2][7];
    }

    // main method
    public static void main(String[] args) {
        CardGame game = new CardGame();
        game.createDeck();
        game.dealCards();
        System.out.println("1st hand's: " + game.handPoints(0));
        System.out.println("2nd hand's: " + game.handPoints(1));
    } // main

    // Creates a new deck of cards
    public void createDeck() {
        String s;
        char f;
        for (int i=0; i<4; i++) {
            s = intToSuit(i+1);
            for (int j=0; j<13; j++) {
                f = intToFace(j+1);
                cards[13*i+j] = new Card(s, f);
            }
        }
    } // createDeck

    // Converts integer to suit
    private static String intToSuit(int v) {
        // code omitted for brevity
    } // intToSuit

    // Converts integer to face value
    private static char intToFace(int v) {
        // code omitted for brevity
    } // intToFace
```
26. (cont…)  
You are to complete these 2 methods in CardGame.java:

(a) **dealCards()** method to deal (give out) 7 cards to each of the 2 hands. (A player is called a hand in a card game.)

Each card is to be dealt in this way. A random integer, \( \text{rand} \), between 0 and 51 (inclusive) is generated and \( \text{cards[\text{rand}]} \) is then assigned to the hand.

Obviously there must not be duplicate cards among the 14 cards that are dealt. That is to say, if a card has already been dealt to a hand, the same card cannot be dealt again. [15 marks]

(b) **handPoints(int hand)** method where **hand** is either 0 or 1, representing one of the two hands. This method is to compute the total points of the 7 cards in a hand. The point system is as follows: 1 point for an ‘Ace’, 2 to 10 points for ‘2’ to ‘10’ respectively, 11 points for a ‘Jack’, 12 points for a ‘Queen’, and 13 points for a ‘King’. [8 marks]