Solution to Midterm 2005

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

a) (X < 5) OR (Y[Y[1]] = 8)

b) (Y[4] ÷ 2 = X) AND (X MOD 2 = Y[1] MOD 2)

c) (X > Y[2]) AND (X ≤ Y[3])

d) Y[Y[0]] × 2 = (2 × X + X MOD 3)

e) X × X ≤ Y[1] × Y[2]

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

a) (X < 5) OR (Y[Y[1]] = 8)

F OR Y[4] = 8

F OR 10 = 8

F OR F

F

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

b) (Y[4] ÷ 2 = X) AND (X MOD 2 = Y[1] MOD 2)

10 ÷ 2 = 5 AND (5 MOD 2 = 4 MOD 2)

5 = 5 AND (1 = 0)

T AND F

F

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

c) (X > Y[2]) AND (X ≤ Y[3])

5 > 6 AND 5 ≤ 8

F AND T

F

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

d) Y[Y[0]] × 2 = (2 × X + X MOD 3)

Y[2] × 2 = (2 × 5 + 5 MOD 3)

6 × 2 = (10 + 2)

12 = 12

T

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

e) $X \times X \leq Y[1] \times Y[2]$

$5 \times 5 \leq 4 \times 6$

$25 \leq 24$

F

Question 1a)

- The integer X has the value 5, and the array Y initially contains the following values: {2, 4, 6, 8, 10}. Only one of the following Boolean expressions evaluates to true. Circle the letter corresponding to the expression that evaluates to true.

a) (X < 5) OR (Y[Y[1]] = 8)

b) (Y[4] ÷ 2 = X) AND (X MOD 2 = Y[1] MOD 2)

c) (X > Y[2]) AND (X ≤ Y[3])

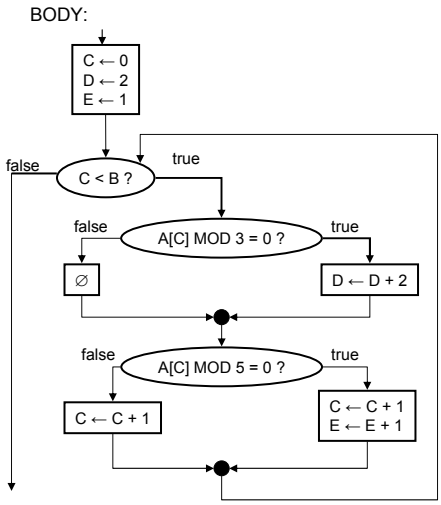
d) Y[Y[0]] × 2 = (2 × X + X MOD 3)

e) $X \times X \leq Y[1] \times Y[2]$

Question 1b)

What would be the values of D and E after the following algorithm executes, if A has the value { 6, 15, 8, 20, 14, 10 } and B has the value 6?

GIVENS: A, B
RESULTS: D, E
HEADER (D,E) ← AnAlgorithm(A, B)
INTERMEDIATE: C



Trace of algorithm

	A	B	C	D	E
Initial values	{ 6, 15, 8, 20, 14, 10 }	6	?	?	?
C ← 0			0		
D ← 2				2	
E ← 1					1
C < B : true					
A[C] MOD 3 = 0 : true					
D ← D + 2				4	
A[C] MOD 5 = 0 : false					
C ← C + 1			1		
C < B : true					
A[C] MOD 3 = 0 : true					

Trace of algorithm

	A	B	C	D	E
from previous page	{ 6, 15, 8, 20, 14, 10 }	6	1	4	1
D ← D + 2				6	
A[C] MOD 5 = 0 : true					
C ← C + 1			2		
E ← E + 1					2
C < B : true					
A[C] MOD 3 = 0 : false					
∅					
A[C] MOD 5 = 0 : false					
C ← C + 1			3		
C < B : true					
A[C] MOD 3 = 0 : false					

Trace of algorithm

	A	B	C	D	E
from previous page	{ 6, 15, 8, 20, 14, 10 }	6	3	6	2
∅					
A[C] MOD 5 = 0 : true					
C ← C + 1			4		
E ← E + 1					3
C < B : true					
A[C] MOD 3 = 0 : false					
∅					
A[C] MOD 5 = 0 : false					
C ← C + 1			5		
C < B : true					
A[C] MOD 3 = 0 : false					

Trace of algorithm

	A	B	C	D	E
from previous page	{ 6, 15, 8, 20, 14, 10 }	6	5	6	3
∅					
A[C] MOD 5 = 0 : true					
C ← C + 1			6		
E ← E + 1					4
C < B : false					

Answer: D = 6, E = 4

Question 2a)

- The following program attempts to extract a sub-array from an array input by the user. The new **`subArray`** contains only the elements of the input array that have odd indices (**`anArray[1], anArray[3]`**…). However, two errors were made when producing this program. One error is a syntax error, while the other error is a logic error in the algorithm.
- The first error is that there were **3** errors in the program ☹.
  - Software engineering advice: Changes made at the last minute are often the most error-prone.

```
public static void main ( String[] args )
   {
      // DECLARE VARIABLES / DATA DICTIONARY
      int[] anArray;        // GIVEN: an array of values
      int anArrayLength;    // GIVEN: number of values in anArray
      int subArrayLength;   // INTERMEDIATE: size of a sub-array
      int index;            // INTERMEDIATE: position in an array
      int[] subArray;       // RESULT: new sub-array
      // READ IN GIVENS
      System.out.print( "Enter an array of integers: " );
      anArray = ITI1220.readIntLine( );
      anArrayLength = anArray.length;
      // ALGORITHM BODY
      subArrayLength = anArrayLength / 2;
      subArray = int[subArrayLength];       // create array
      index = 0;

      while (index < anArrayLength)
      {
         subArray[index] = anArray[index*2+1];
         index = index + 2;
      }
      // PRINT RESULTS AND MODIFIEDS
      index = 0;
      while (index < subArrayLength)
      {
         System.out.print(subArray[index] + " ");
         index = index + 1;
      }
   }
```

syntax: missing **new**

logic: **index < subArrayLength**

logic: add 1, not 2

Question 2b)

```
int [] x;
int r;
int i;

/* 1 */ x = new int[] {2, 3, 4, 5, 6};
/* 2 */ r = 1;
/* 3 */ i = x.length-1;
/* 4 */ while ( i > 0 )
{
   /* 5 */ r = r * x[i];
   /* 6 */ if (r > 100)
   {
      /* 7 */ i = i - 1;
   }
   else
   {
      /* 8 */; // do nothing
   }
   /* 9 */ i = i - 1;
}
/* 10 */ System.out.println( r );
```

What does this program print?

a) 720
b) 360
c) 144
d) 120
e) 60
f) There is a run-time error

Trace of program

#		x	r	i
	Initial values	?	?	?
1	x = new int[] {2, 3, 4, 5, 6};	{2, 3, 4, 5, 6}		
2	r = 1		1	
3	i = x.length-1			4
4	while (i > 0) : true			
5	r = r * x[i]		6	
6	if (r > 100) : false			
8	; // do nothing			
9	i = i - 1			3
4	while (i > 0) : true			
5	r = r * x[i]		30	

Trace of program

#		x	r	i
	From previous page	{2, 3, 4, 5, 6}	30	3
6	if (r > 100) : false			
8	; // do nothing			
9	i = i - 1			2
4	while (i > 0) : true			
5	r = r * x[i]		120	
6	if (r > 100) : true			
7	i = i - 1			1
9	i = i - 1			0
4	while (i > 0) : false			
10	System.out.println(r)			

Question 2b)

```
int [] x;
int r;
int i;

/* 1 */ x = new int[] {2, 3, 4, 5, 6};
/* 2 */ r = 1;
/* 3 */ i = x.length-1;
/* 4 */ while ( i > 0 )
{
   /* 5 */ r = r * x[i];
   /* 6 */ if (r > 100)
   {
      /* 7 */ i = i - 1;
   }
   else
   {
      /* 8 */; // do nothing
   }
   /* 9 */ i = i - 1;
}
/* 10 */ System.out.println( r );
```

What does this program print?

a) 720
b) 360
c) 144
d) 120
e) 60
f) There is a run-time error

Question 3) Translate to Java

GIVENS:	A	(An array of integers)
	N	(Number of values in the array A)
RESULT:	Duplicates	(Boolean: True if A contains duplicate values)
HEADER:	Duplicates ← HasDuplicates(A, N)	
INTERMEDIATES:	CheckIndex	(Index of left comparison value)
	DupIndex	(Index of right comparison value)

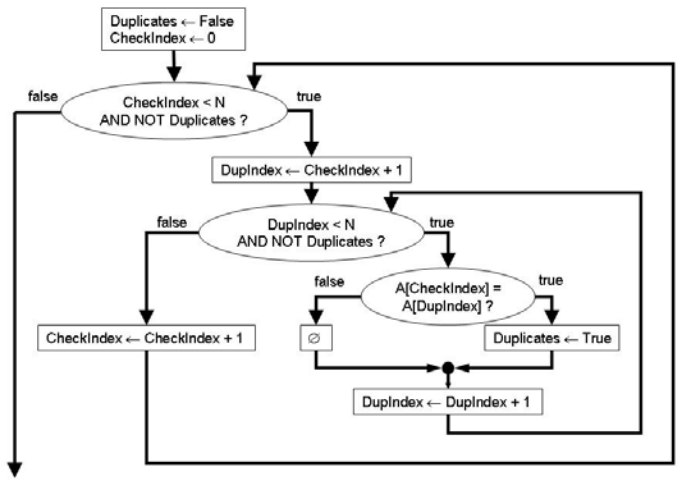
Question 3 (first part)

```
public static void main( String[] args )
 {
   // DECLARE VARIABLES / DATA DICTIONARY

   int [] a;           // GIVEN: array of integers
   int n;              // GIVEN: number of values in the
                       // array A
   boolean duplicates; // RESULT: true if a contains
                       // duplicate values
   int checkIndex;     // INTERMEDIATE: index of left
                       // comparison value
   int dupIndex;       // INTERMEDIATE: index of right
                       // comparison value
```

Question 3 (second part)

BODY:



Question 3 (second part)

```
// BODY OF ALGORITHM
  duplicates = false;
  checkIndex = 0;
  while ( checkIndex < n && ! duplicates )
  {
     dupIndex = checkIndex + 1;
     while ( dupIndex < n && ! duplicates )
     {
        if ( a[checkIndex] == a[dupIndex] )
        {
           duplicates = true;
        }
        else
        {
           ; // do nothing
        }
        dupIndex = dupIndex + 1;
     }
     checkIndex = checkIndex + 1;
  }
```

Question 4: Algorithm

- Suppose that you have an array BoltSizes containing the measured lengths, in millimetres, of various bolts after they have been manufactured. There are N values in this array. The bolts are intended to have the measured length CorrectSize, but the size is permitted to vary, plus or minus, up to and including the amount Tolerance to be acceptable
- Design an algorithm to count the number of bolts that should be rejected.
- For example, if the bolt sizes are 50.03, 50.02, 49.97, 50.04, 50.01, 50.00, and 49.99, the correct size is 50.0, and the tolerance is 0.02, the algorithm would return 3.

Question 4)

GIVENS:	BoltSizes	(Array of bolt sizes, in mm)
	N	(Number of values in array)
	CorrectSize	(Correct size for a bolt, in mm)
	Tolerance	(Allowed variance in size, in mm)
RESULT:	Count	(Number of bolts to reject)

HEADER: Count ← CountRejectedBolts (BoltSizes, N, CorrectSize, Tolerance)

INTERMEDIATES:

	Index	(Index for the array)

Question 4)

BODY: