UNIVERSITY OF TWENTE.

Exam Application Development

Bachelor 1 *Industrial Design, ET*

Module/code: Module 4, 2017-201700366-2B
Date: July, 2th, 2018 08:45
Time: 2.5h *(+25% for students entitled to extra time)*
Module-coordinators: W. Dankers, M.P. Zwier
Instructor: F. van Slooten

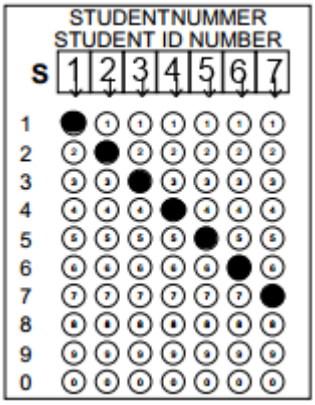
Type of exam: closed book, multiple choice + open questions

Appendices (2): Multiple Choice form, API documentation

Permitted aids: none

Indications:

The use of book, notes, calculator etc. is not allowed during this exam. Enter the multiple-choice answers on the attached Multiple-Choice Form. You can write the answers to the open questions on the additional examination paper. Write name, initials, student number and subject name on the Multiple-Choice Form and exam paper. Make sure that you fill in the student number correctly and tick the corresponding boxes on the answer sheet. For example, if the student number would be s1234567 as shown on the right.
Try to answer open questions briefly (possibly explained with sketches).



This exam consists of:
- Questions 1 to 16 on 6 pages
- An appendix with API documentation of 6 pages.

The Java code presented in the questions is working code, which has been tested and executed in Eclipse. For multiple-choice questions, the answers may display code that is incorrect.

Rating: question 1 to 10: 2 points, 11 to 16: 4 points (total points 44 / 4.4 = grade).

Multiple choice questions

1. If you would like to store an analog signal value which is read from an electrical circuit and save the associated value in a variable, what type of variable would be suitable to store this value?

A. int
B. String
C. object
D. boolean

2. Suppose we connect a LED to pin 8 of an Arduino. What does the code below do?

```
void setup() {
  pinMode(8, OUTPUT);
}
```

A. This sets up the LED and turns it off
B. Turns the LED off by writing LOW out on the pin
C. This sets up the pin that the LED is connected to as an OUTPUT pin
D. This sets up the pin that the LED is connected to as an INPUT pin

3. See the definition of the class Ball. Which word can be filled in at the spot marked with /*xxx*/?

A. `void`
B. `int`
C. `public`
D. `Graphics`

4. See the definition of the class Ball. If we would like to give the Ball a random size between 10 and 20, which line of code would we have to add to the constructor of the class Ball.

If necessary, use the API documentation of the Random class in the appendix.

A. `size = rg.nextInt(20);`
B. `size = 10 + rg.nextInt(10);`
C. `size = Random(10,20);`
D. `size = rg.nextInt(10,20);`

```
public class Ball {
  private int size = 15;
  private int x,  y, dx = 2, dy = 2;
  private int panelWidth, panelHeight;
  private Color color;

  public Ball(int panelWidth, int panelHeight) {
    this.panelWidth = panelWidth;
    this.panelHeight = panelHeight;

    Random rg = new Random();
    x = rg.nextInt(panelWidth);
    y = rg.nextInt(panelHeight);

    int r = rg.nextInt(256), g = rg.nextInt(256),
    b = rg.nextInt(256);
    color = new Color(r, g, b);
  }

 public /*xxx*/ paintComponent(Graphics g) {
    g.setColor(color);
    g.fillOval(x, y, size, size);
  }
}
```

5. What would be a correct order of the lines of code below to have a robot make a point turn (turn around its own axis) with a given angle?

```
1: evshield.bank_a.motorRunDegrees(SH_Motor_2, SH_Direction_Reverse, speed,
degrees, SH_Completion_Wait_For, SH_Next_Action_Float);
2: evshield.bank_a.motorRunDegrees(SH_Motor_1, SH_Direction_Forward, speed,
degrees, SH_Completion_Dont_Wait, SH_Next_Action_Float);
3: unsigned int degrees = abs(angle) * (TRACKWIDTH / WHEEL_DIAM);
```

A. 1, 2, 3
B. 3, 2, 1
C. 3, 1, 2
D. 1, 3, 2

6. Which of the loop statements would generate the following output?:

```
2 6 18 54
```

A. `int n=3; while (n < 100) { System.out.print( " "+n ); n=n*2; }`
B. `for (int n=54; n > 0; n=n/3) System.out.print( " "+n );`
C. `for (int n=2; n < 54; n=n*3) System.out.print( " "+n );`
D. `int n=2; while (n < 100) { System.out.print( " "+n ); n=n*3; }`

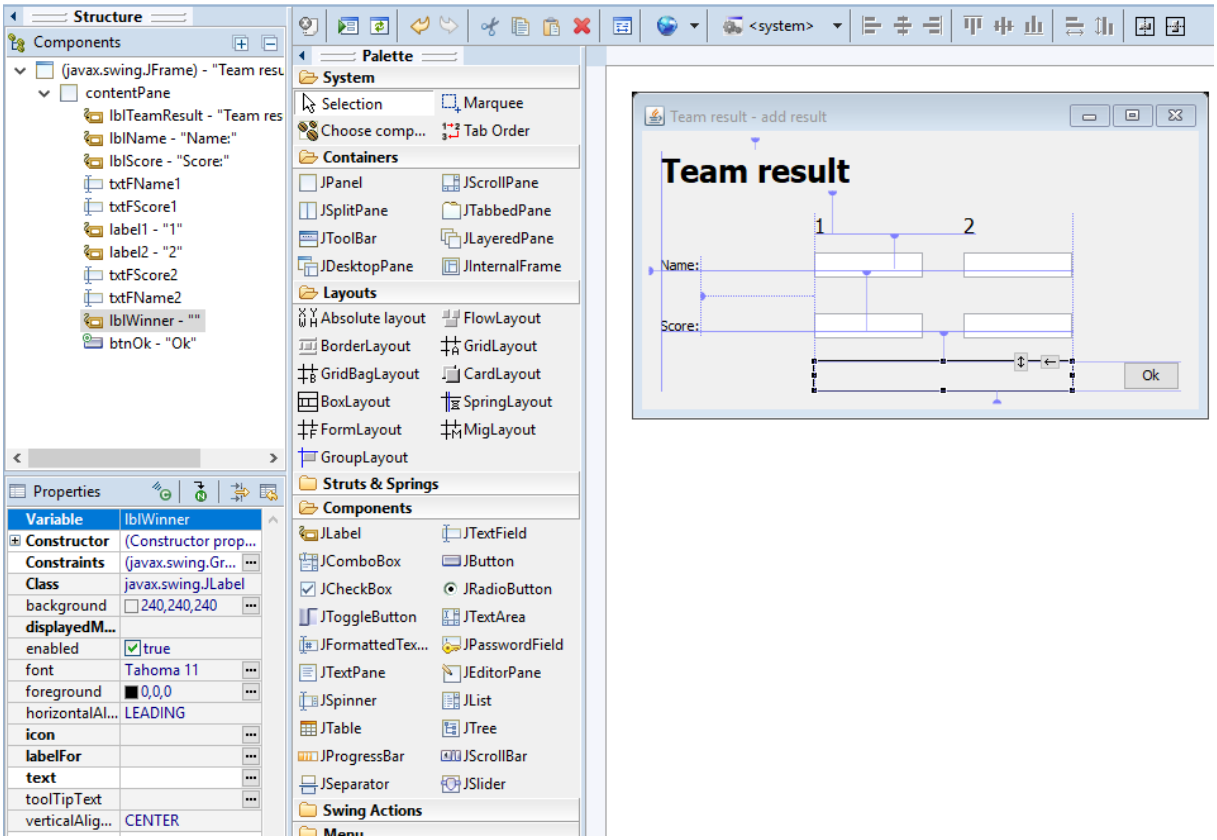
7. At the next page, a definition of a user interface in Eclipse is given. What is the correct declaration and initialization of the selected component?

A. `JLabel lblWinner = new JLabel("");`
B. `lblWinner = new JLabel("Winner: ");`
C. `lblWinner = { JLabel("") };`
D. `JLabel lblWinner; new JLabel("Winner: ");`

8. See the definition of a user interface on the next page. Pressing the "Ok" button (`btnOk`) will read values from the text fields. Shown are lines of code that could be placed in the method that handles the button pressed event.
Which line will get the score of team one and convert that to an integer?

If necessary, use the API documentation of the classes JTextfield, Integer and String in the appendix.

A. `int s1 = txtFScore1.intValue();`
B. `int s1 = Integer.parseInt( txtFScore1.setText() );`
C. `int s1 = Integer.parseInt( txtFScore1.getText() );`
D. `int s1 = parseInt( txtFScore1.getText() );`



9. Integer variables `s1` and `s2` contain the scores of the two teams. A method with the name "result" will display the result. Its parameter indicates the winning team, or a 0 for a draw (no winner). Which code can be used to properly display the result?

A.
```
if (s1>s2) result(1);
if (s2>s1) result(2);
else result(0);
```

B.
```
if (s1>s2) result(1);
else if (s2>s1) result(2);
else result(0);
```

C.
```
if (s1=s2) result(0);
if (s1>s2) result(1);
if (s2>s1) result(2);
```

D.
```
if (s1==s2) result(0);
if (s1>s2) result(1);
else result(2);
```

10. What would be a proper implementation of a new method for the Arduino/C++ class Explorer which determines if the touch sensor is pressed? The `isPressed()` method of class EVs_NXTTouch returns a value of type `bool`.

You can find the declaration of the class Explorer in the appendix (Explorer.h).

A.
```
public bool getTouch() {
  return touch.isPressed();
}
```

B.
```
bool Explorer::getTouch(int pressed) {
  return touch.isPressed();
}
```

C.
```
public void getTouch() {
  touch->isPressed();
}
```

D.
```
bool Explorer::getTouch() {
  return touch->isPressed();
}
```

Open questions

11. Given is the class `DrawingPanel` below. Write the code for the `paintComponent()` method that draws a filled red circle with a diameter of 300 on the screen. The code is placed where the comment `// ...` is now.

If necessary, use the API documentation from the Graphics class in the appendix.

```
public class DrawingPanel extends JPanel {
  @Override
  public void paintComponent(Graphics g) {
     // ...
  }
}
```

For questions 12-15 you will have to write out code for a class and some methods. You may choose to use Java syntax or Arduino/C++ syntax for this. Please make it clear which syntax you choose at question 12 and stick to that syntax for all questions (12-15).

12. Write code for a class Drink, which is used in an application for a restaurant. The class realizes the name, the size (a whole number, which indicates the amount of milli-liter) and a table number (the table from which the drink was ordered).

13. Write a method for the class Drink which can set the table number to a given value.
The value is a parameter of the method. For example, the method can be used when the waiter enters a new order.

14. Write a complete method for the Drink class that prints the information (values of all class-variables) to the Console (eg. using `System.out.print()` or `System.out.println()`), or to the Serial Monitor (using `Serial.print()` or `Serial.println()`).

15. See the previous 3 questions. Write a method named "addDrinks" which:
- Adds 2 new drinks to a list of drinks made with an ArrayList
- Calls method *showDrinks()* to show all drinks in the list (you do not have to write this method yourself, just show a call to it).

If necessary, use the API documentation of the ArrayList class in the appendix. For Arduino/C++ you may assume that the class ArrayList is available in the Arduino environment.

16. Sketch a design for a user interface which allows to input drinks for a table. You may assume you would have to create the userinterface using the Window Builder in Eclipse.

The user interface must contain the following components:
- An element to input or select the table number
- Element(s) to enter the name of the drink or select the drink
- A button to add a drink
- A large 'display' (eg. panel or list) in which the drinks ordered will be visible
- A button to submit the order

Add a list of the types of user interface components that you have used. You can use the types as listed in the Palette as displayed on the screen print on page 4.