|  |
| --- |
|   The City School  North Nazimabad Boys Campus            Summer Vacation Worksheet   ICT Grade 8   |

 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Level**  | CSII  | **Module**  | 4  | **Task**  | 3  | **Class**  |
| **Student**  |  |  |  |  |  |  **Date**  |

|  |
| --- |
|  **1. Put a**  **if the sentence is correct**  |
| 1. The control MenuStrip helps us create toolbars for a program.  |   |
| 2. The main use of SnapLines is positioning labels and textboxes.  |   |
| 3. A responsive interface is always fast and does not waste the user’s time unnecessarily.  |   |
| 4. Logical errors are very easy to find as the program will inform you with a blue wiggly line.  |   |
| 5. The error list panel only displays design time errors.  |   |
| 6. A runtime error always occurs when the program is running.  |   |
| 7. While you are debugging a program the yellow highlight indicates the next instruction to be executed.  |   |
| 8. While debugging a program, you can go over your program line by line by pressing F11.  |   |
|  **2.Select the correct answer**  |
| The ToolStrip control  | * creates menus for a program

creates toolbars for a program * creates textboxes for a program
 |
| We use the Try….Catch statement in order to  | * find a design time error

debug a program * prevent code from crashing when an error occurs
 |
| When you use the Try…Catch statement we use an **ex** variable which…  | * is an object we have to define beforehand in order to create it
* is an abject and is created automatically when an error occurs
* is the name of a class
 |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  **Level** CSII  | **Module**  | 4  | **Task**  | 2  | **Class**  |
| **Student**  |  |  |  |  |  **Date**  |
| **1. Select the correct answer with reference to Microsoft Small Basic**  |
| The condition number >5 **And** number <10 is true…  |  if number = 15 if number = 10 if number = 7  |
| The condition number >5 **Or** number=10 is false…  | * if number =4
* if number=10

if number =11  |
| The meaning of the logical operator **Xor**is that…  | one side or another must be true both sides must be true  one side or another must be true but not both  |
|

|  |
| --- |
| **Do while** condition  instructions **Loop**  |

In the loop :  | the instructions are executed at least one time the instructions are executed while the condition remains true the instructions are executed until the condition becomes true  |
|

|  |
| --- |
| **Do**   instructions **Loop Until**condition |

In the loop :  | the instructions are executed while the condition remains true the instructions are executed until the condition becomes True  the instructions are executed at least one time  |
|

|  |
| --- |
| **Do Until** condition instructions **Loop**  |

In the loop :  | * the instructions are executed while the condition remains false
* the instructions are executed until the condition becomes false

the instructions are executed while the condition remains true  |

|  |
| --- |
|  **2. Put a**  **if the sentence is correct**  |
| 9. The conditional statement **If…Then** is executed only if the condition is true  |   |
| 10. The conditional statement **If…Then…Else** is executed only if the condition is true.  |   |
| 11. Mouse events are actions that a user can do with a mouse when a program is running  |   |
| 12. A mouse down event is the same as a mouse click  |   |
| 13. In one form you can detect only one kind of mouse event  |   |
| 14. The **For…Next** loop using the keyword **Step**will always get positive numbers  |   |
| 15. When we use the **Do…..Loop While** loop, we can specify how many times the commands inside the loop will be executed  |   |
| 16. When we use the **Do…..Loop Until** condition the instructions inside the loop are executed at least one time  |   |
| 17. The command **user1=name(2)**stores the value of the variable *name2* to a particular place on the *user*array  |   |
| 18. An executed function does not return any value  |   |
| 19. A subroutine does not return a value  |   |
| 20. We pass an argument by value by specifying the **ByVal** keyword for the corresponding parameter in the procedure definition  |   |
| 21. We create modules in order to keep code files clean and easy to read and maintain  |   |
| 22. When we move a sub into a module we have to declare this sub as public in order for this procedure to be accessible from inside your other files  |   |

 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  **Level** CSII  | **Module**  | 4  | **Task**  | 1  | **Class**  |
| **Student**  |  |  |  |  |  **Date**  |
| **1.Select the correct answer**  |
| 1. We use Windows forms application.  | * to create a command line application.
* to create a VB class library.
* to create an application with a Windows user interface.
 |
| 2. On the properties panel, the Text property  | * defines the title of the selected element.
* defines a name with which we can refer to the selected element.
* defines if the selected element will be usable or disabled.
 |
| 3. On the properties panel, the Name property  | * defines the role that will be reported to accessibility clients.
* defines a name that will be reported to accessibility clients.
* defines a name with which we can refer to the selected element.
 |
| 4. On the properties panel, the Enabled property  | * defines whether the control can accept data that the user drags onto it.
* defines if the selected element will be usable or disabled.
* defines the title of the selected element.
 |
| 5. The common control **button**  | * allows you to display information like a text field.
* allows you to specify program instructions to be run when a user clicks the button.
* enables the user to enter text and provides multiline editing.
 |
| 6. The common control **label**  | * allows you to display a list of items with a check box on the left side of each item.
* allows you to display information like a text field for the results of your program’s actions.
* allows you to select or clear the associated option.
 |
| 7. When a variable has been declared as Boolean, it can have  | text as value. numbers as value. the values **True** or **False**.  |
| 8. When a variable has been declared as String, it can have  | numbers as value. text as value.  integer numbers as value.  |
| 9. We declare a variable as **Public** in order  | to be visible and accessible to one particular code block. to be visible and accessible to many code blocks.  to not change its value in any code block.  |

|  |  |
| --- | --- |
| **1.**  | **Put a** **if the sentence is correct**  |

1. The Edison robot can only detect obstacles that are close to it. 



1. In event driven programming the flow of the program depends on

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Level**  | CSII  | **Module**  | 5  | **Task**  | 2  | **Class**  |
| **Student**  |  |  |  |  |  | **Date**  |

 incoming events. 



1. The event handler has two parameters: the event that will happen and

 the function that will be called when the event happens. 



1. With the event handler you can check the function’s code once and

 make sure that it works as you planned. 



1. You can only write more complicated programs by using “if”, “elif” and

 “else” statements. 



1. When you use “if”, “elif” and “else” statements in the Python programming language, it means that you have to set three conditions. 



1. You can use the Ed.ReadObstacleDetection() function in order to clear

 any unwanted detections. 

|  |  |  |
| --- | --- | --- |
|  | **2. Match the correct answer**  |  |
| **1**  | whileEd.ReadObstacleDetection()!=Ed.OBSTACLE\_AHEAD: pass   | O  |   | O  | a. This program makes the robot call an event when it detects an obstacle.  |
| **2**  | Ed.ObstacleDetectionBeam(Ed.ON) Ed.RegisterEventHandler(Ed.EVENT\_OBSTACLE\_AHEAD,"avoidObstacle") while True: Ed.Drive(Ed.FORWARD,Ed.SPEED\_3,Ed.DISTANCE\_UNLIMITED) defavoidObstacle(): Ed.Drive(Ed.STOP,Ed.SPEED\_1, 0) Ed.ReadObstacleDetection()  | O  |   | O  | b. This program has different paths that the robot can take when it detects an obstacle.  |
| **3**  | Ed.ObstacleDetectionBeam(Ed.ON) Ed.Drive(Ed.FORWARD,Ed.SPEED\_10, Ed.DISTANCE\_UNLIMITED) while True: ifEd.ReadObstacleDetection()==Ed.OBSTACLE\_AHEAD: Ed.Drive(Ed.FORWARD, Ed.SPEED\_4, 90) else: Ed.Drive(Ed.FORWARD,Ed.SPEED\_4, Ed.DISTANCE\_UNLIMITED)  | O  |   | O  | c. This programtells Edison robot to react when it detects obstacles ahead.   |

|  |  |
| --- | --- |
| **1.**  | **Put a** **if the sentence is correct**  |

1. EdPy is a text-based programming language based on Python. 



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Level**  | CSII  | **Module**  | 5  | **Task**  | 1  | **Class**  |
| **Student**  |  |  |  |  |  | **Date**  |

1. To start a new program, you need to start typing in line 1. 



1. In the Compiler output area, you can see if the program has any errors. 



1. To move the robot,you need to use the drive functionthat has two

 parameters, the direction and the distance. 



1. Always separate the function parameters with a full stop. 



1. The single-line comments in Python are created by putting the hash

 character at thebeginning of aline. 



1. With the For loop you know how many times the commands will be

 executed. 

|  |
| --- |
| **2. Match the correct answer**  |
| **1**  | degreesToTurn=180  | O  |   | O  | d. This program will make the robot drive forward with a speed of 2 and when a clap is detected will play a beep sound and will drive backwards with a speed of 2.  |
| **2**  | Ed.LeftLed(Ed.On) Ed.Timewait(10, Ed.TIME\_SECONDS)  | O  |   | O  | e. Setting this variable, you make the robot turn so as to place horizontally the base of the triangle.  |
| **3**  | Ed.Drive(Ed.FORWARD,Ed.SPEED\_2,10) Ed.Timewait(2,Ed.TIME\_SECONDS) Ed.ReadClapSensor() whileEd.ReadClapSensor( )==Ed.CLAP\_DETECTED: pass Ed.PlayBeep() Ed.Drive(Ed.BACKWARD,Ed.SPEED\_2,6)   | O  |   | O  | f. This programswitches on the left LED of the robot for 10 seconds.   |
| **4**  | degreesToTurn=120 degreesToTurn2=30  | O  |   | O  | g. This program makes the robot turn 180 degrees to the left.  |

|  |  |
| --- | --- |
| **1.**  | **Put a** **if the sentence is correct**  |

1. The Raspberry Pi can operate like a desktop computer. 



1. The Raspberry Pi can use a Micro USB, an HDMI, a SD card and a CPU. 



1. GPIO pins can only be used as outputs. 



1. In the Raspbian environment you can only program in the Scratch

 language. 



1. Raspberry Pi supports all math operations. 



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Level**  | CSII  | **Module**  | 5  | **Task**  | 3  | **Class**  |
| **Student**  |  |  |  |  |  | **Date**  |

1. We can import existing code into the Raspberry Pi. 



1. Python input function is ( ) 



1. To open a code In Python you click new file 



1. GPIO stands for General information and output 



1. To get integer numbers we type**" int ( input() )".** 

1. In Raspberry pi we use the Micro USB port for Power Supply. 

1. SD memory card is a storage device In Raspberry pi. 

1. GPIO Pins cannot be used to make LED light on. 

1. For decimal input we type **"float ( input() )".** 

1. Internet cannot be connected in in Raspberry pi 