

Plainview Old Bethpage Central School District

Computer Programming I Curriculum

Jill Tretola
Summer 2013

Computer Programming I Curriculum

Visual Basic

Resources:

Beth B., Bruce P. (1999). An Introduction to Programming Using Microsoft Visual Basic. Lawrenceville Press.

Sprague, M., Phillips, A. & Nield, D. (2000). Microsoft Visual Basic Introduction to Programming 6.0. Cincinnati, Ohio: South-Western Educational Publishing Thomson Learning.

Sprague, M., Phillips, A. & Vermeer, P. (2000). Microsoft Visual Basic Introduction to Programming 6.0. Activities Workbook. Cincinnati, Ohio: South-Western Educational Publishing Thomson Learning.

Thompson, A. (1998). Microsoft Visual Basic Projects for the Classroom. Academic Cooperative Microsoft Corporation.

Pacing Chart	
Lesson 1: 1.5 weeks	Lesson 2: 1.5 weeks
Lesson 3: 3 weeks	Lesson 4: 5 weeks
Midterm Review: 1 week	
Lesson 5: 4 weeks	Lesson 6: 3 weeks
Lesson 7: 3.5 weeks	Lesson 8: 3 weeks
Culminating Projects: 8 weeks	
Final Project: 1.5 weeks	
Final Exam Review 1 week	
Time has been left to allow for tests and quizzes as well as review for each.	

Lesson 1: The Basics		
Processing Skills	Objectives	Essential Vocabulary
1) Read for information and understanding. 2) Understand cause and effect relationship of an event driven program. 3) Preplan layout of form. 4) Listen and take notes.	<ul style="list-style-type: none"> • Identify and use the basic tools in the Visual Basic Integrated Development Environment. • Create a form and add objects to it. • Explain the meaning of “event driven”. • Program an event procedure. 	<ul style="list-style-type: none"> • Graphical User Interface(GUI) • Menu • Event-driven • Object • Integrated Development Environment • Controls • Tooltip • Dynamic Link Libraries • Forms • File extension • Event procedure
Suggested Activities: Textbook Application Problems Page 23 Levels 1 and 2 Workbook Page 19 Application Program		
Suggested Homework Assignments: Workbook pages 1-8		

Lesson 2: Forms		
Processing Skills	Objectives	Essential Vocabulary
1) Read for information and understanding. 2) Understand cause and effect relationship of an event driven program. 3) Preplan layout of form. 4) Listen and take notes. 5) Preplan and organize an outline to be used for code in subroutine.	<ul style="list-style-type: none"> • Customize the look and feel of a form. • Develop standard forms that can be reused in other application. • Create forms that react to user events like clicking a button and pressing a key. • Use variables and pass information from one part of the project to another. • Import graphics to a form. 	<ul style="list-style-type: none"> • Syntax • Keyword • End statement • Twip • Load • Unload • Assignment statement • Variable • Constant • Data type • Modules
Suggested Activities: Textbook Application Problems Page 45 Levels 1 and 2		
Suggested Homework Assignments: Workbook pages 11-18		

Lesson 3: Menus and MDIs

Processing Skills	Objectives	Essential Vocabulary
<p>1) Read for information and understanding.</p> <p>2) Understand cause and effect relationship of an event driven program.</p> <p>3) Preplan layout of form.</p> <p>4) Listen and take notes.</p> <p>5) Preplan and organize an outline to be used for code in subroutine.</p> <p>6) Preplan menu layout to maximize user friendliness.</p>	<ul style="list-style-type: none">• Design a menu bar with menus and submenus.• Create an operational application.• Plan and document an application.• Add forms from other projects.• Use MDI forms.	<ul style="list-style-type: none">• Access keys• Shortcut keys• Separator bar• Menu Editor• Multiple Document Interface (MDI)• Personal library• Cascade• Tile

Suggested Activities: Textbook Application Problems Pages 62-63 Levels 1 and 2
Workbook Page 49 Application Program

Suggested Homework Assignments: Workbook pages 21-30

Lesson 4: Decisions, Looping Arrays, and Subroutines

Processing Skills	Objectives	Essential Vocabulary
<p>1) Read for information and understanding.</p> <p>2) Understand cause and effect relationship of an event driven program.</p> <p>3) Preplan layout of form.</p> <p>4) Listen and take notes.</p> <p>5) Preplan and organize an outline to be used for code in a subroutine.</p> <p>6) Understand conditional statements as it pertains to if-then-else statements.</p> <p>7) Understand consequences as it pertains to Do-While and Do-</p>	<ul style="list-style-type: none"> • Plan the solution to a programming problem. • Understand and apply the rules of logic to True/False expressions. • Write <i>If..Then..Else..End If</i> statements. • Write <i>Do While</i> and <i>Do Until</i> statements. • Write For loops. • Create program loops that depend on logical expressions. • Declare and use arrays. • Use built in functions for a variety of tasks. • Determine when to use a definite loop and when to use an indefinite loop. 	<ul style="list-style-type: none"> • User interface • Algorithm • Stepwise refinement • Subroutine • Event procedure • Data structure • Array • ListIndex • File extension • Syntax • Boolean or logical expression • Debug • Program flow • Relational operators • Assignment operator • ASCII code • Definite loop • Indefinite loop • Increment • Body • Sentinel • Mod • Menu control array • Index or subscript

<p>Until loops. 8) Cold read, plan and implement programming projects.</p>		<ul style="list-style-type: none"> • Scope • Visibility • Lifetime • Private variable • General Declarations • Linear search • Flag • Concatenation • Error trapping
<p>Suggested Activities: Textbook Application Problems Pages 93-94 Levels 1 and 2 Workbook Page 55 Application Program</p>		
<p>Suggested Homework Assignments: Workbook pages 33-48</p>		

Lesson 5: Improving the User Interface

Processing Skills	Objectives	Essential Vocabulary
<p>1) Read for information and understanding.</p> <p>2) Understand cause and effect relationship of an event driven program.</p> <p>3) Preplan layout of form.</p> <p>4) Listen and take notes.</p> <p>5) Preplan and organize an outline to be used for code in subroutine.</p> <p>6) Cold read, plan and implement programming projects.</p>	<ul style="list-style-type: none"> • Use several variations of the <i>MsgBox</i> and <i>InputDialog</i> dialog boxes. • Understand how to use Visual Basic built in constants. • Implement pop-up menus. • Add a toolbar to your project. • Use variations of the Common Dialog control to improve your program's user interface. 	<ul style="list-style-type: none"> • MsgBox function • Modal • InputBox function • Null string • Prompt • Run-time error • Pop-up menu • Button • ImageList control • Random access file • Common Dialog control • Open statement • FreeFile function • Print # statement • Close • Button Collection • Property pages • Sequential file • Input function • LOF()
<p>Suggested Activities: Textbook Application Problems Pages 128-130 Levels 1 and 2</p>		
<p>Suggested Homework Assignments: Workbook pages 51-64</p>		

Lesson 6: Database Programming

Processing Skills	Objectives	Essential Vocabulary
<p>1) Read for information and understanding.</p> <p>2) Understand cause and effect relationship of an event driven program.</p> <p>3) Preplan layout of form.</p> <p>4) Listen and take notes.</p> <p>5) Preplan and organize an outline to be used for code in subroutine.</p> <p>6) Read, understand and use a database.</p>	<ul style="list-style-type: none"> • Explain what a database is and how to use it. • Create a database using Visual Data Manager. • Link a database to a Visual Basic Project. • Use a Visual Basic Program to add and change records in a database. • Link to external databases. • Distinguish between the three kinds of modules: Standard, Class and Form. • Use User Defined Data Types. 	<ul style="list-style-type: none"> • Database • Record • Field • Jet database engine • VisData • Indexed fields • Snapshot, Recordset, Dynaset • Data control • DataSource property • DataField property • MSChart control • ChartData property • Two-dimensional array • Active X Data Objects • Profile • User Defined Data Types(UDT) • Standard module • Class module • Fixed-length string • Variable-length string • Binary file
<p>Suggested Activities: Textbook Application Problems Pages 162-163 Levels 1 and 2</p>		
<p>Suggested Homework Assignments: Workbook pages 67-77</p>		

Lesson 7: Under the Hood

Processing Skills	Objectives	Essential Vocabulary
<p>1) Read for information and understanding.</p> <p>2) Understand cause and effect relationship of an event driven program.</p> <p>3) Listen and take notes.</p> <p>4) Preplan and organize an outline to be used for code and form layout in subroutine.</p> <p>5) Cold read, plan and implement programming projects.</p> <p>6) Decide when it is appropriate to use a definite loop or an indefinite loop.</p>	<ul style="list-style-type: none"> • Use breakpoints and Watch windows to locate program errors. • Use the Immediate window to check and change values. • Check a program's logic by stepping through the program one line at a time. • Use the On Error statements to trap run-time errors. • Create a report using an ODBC driver. 	<ul style="list-style-type: none"> • Bugs • Error trapping • InterCap naming convention • Option Explicit • Syntax errors • Breakpoint • Break mode • Watch expression • Watch Window • Step Into • Error Handler • Err and Error • Err object • Error log or log file • App object • ODBC Driver • Data source name(DSN)

Suggested Activities: Textbook Application Problems Pages 188-189 Levels 1 and 2

Suggested Homework Assignments: Workbook pages 79-87

Lesson 8: What's the Object? Programming With Class

	Objectives	Essential Vocabulary
<p>1) Read for information and understanding.</p> <p>2) Understand cause and effect relationship of an event driven program.</p> <p>3) Preplan layout of form.</p> <p>4) Listen and take notes.</p> <p>5) Preplan and organize an outline to be used for code in subroutine.</p> <p>6) Cold read, plan and implement programming projects.</p> <p>7) Understand variable use when working with multiple classes.</p>	<ul style="list-style-type: none">• Use the Object Browser to explore classes and objects.• Design and Create class modules.• Use class modules, events, and properties to create classes• Create an ActiveX DLL to distribute a class to other applications.	<ul style="list-style-type: none">• Platforms• Reusable code• Cross-platform• Object Oriented Programming(OOP)• Object• Data Encapsulation• Inheritance• Polymorphism• Messages• Data hiding• Operator overloading• Object Browser• Visual Basic for Applications(VBA)• Class Module• New• Instance• Set• StrConv()• ByVal• Dynamic Link Library(DLL)• ActiveX DLL• Let

		<ul style="list-style-type: none">• Get• References
Suggested Activities: Textbook Application Problems Page 219 Levels 1 and 2		
Suggested Homework Assignments: Workbook pages 89-97		
Assessments		
<p>Full period, written, exams will be given at the end of each lesson. They will consist of multiple choice, matching, fill in and extended response type questions. In addition, each lesson will have alternative assessments assigned in the form of application programming problems that will be graded.</p>		

Culminating Programming Projects

1) Slot Machine Game Application

Create a Slot Machine Game application that acts as a simple slot machine. The user starts with 100 tokens. With each “pull” of the handle, the user loses 1 token and the computer “spins” three wheels, each consisting of the numbers 1, 2, and 3. If all three numbers are 1, the user gets 4 tokens; if all are 2, the user gets 8 tokens; if all are 3, the user gets 12 tokens. You can substitute pictures for numbers, if you wish to enhance and individualize your program. The number of tokens that the user has should be displayed on the form and the result of the spin should be displayed in a message box. The user interface should be designed so that it is easy to use.

2) Game of 21 Application

Create a Game 21 application to simulate a simplified version of the game “21”. A deck of cards numbered 1 through 10 is used and any number can be repeated. The computer starts by dealing you (the user) two randomly picked cards, and deals itself three randomly picked cards that are not revealed until the *Check Scores* button is clicked. You may then draw as many cards as you want, one by one. If both scores are over 21, or if both are equal but under 21, the game is declared a draw. Otherwise, the winner is the one with the highest score less than or equal to 21. If one score is over 21 and the other is 21 or less, the player with 21 or less is declared the winner. The result should be displayed in a message box. The user interface should be designed so that all cards are displayed clearly at the appropriate time and all buttons should appropriately labeled so they are self-documenting.

3) Math Tutor Application

Create a Math Tutor Application that displays math problems by randomly generating two numbers, 1 through 10, and an operator (*, +, -, /), and then allows the user to enter the answers (hint: generate random numbers to represent the operators). The application should be able to check the answer and display a message, display the correct answer, and generate new problems. The Smiley graphic (or another appropriate graphic) should be displayed when the user enters a correct answer and removed when the user clicks New Problem. The application interface should allow the user to enter an answer in a text box on the form and buttons should be clearly labeled. There should also be a display that keeps track of the number of problems that the user gets right and wrong.

4) Mastermind Application

Create a Guess the Blocks application to simulate a modified version of the game Mastermind. In this game, Three different colored blocks are lined up and hidden from the player. The player then tries to guess the colors and the correct order of the blocks. There are 4 colored blocks (red, green, blue, yellow) to choose from. After guessing the color of the three hidden blocks the computer displays how many of the colors are correct and how many of the colors are in the right position. Based on this information the player makes another guess and so on until the player has determined the correct order and color of the hidden blocks. Use R for Red, B for Blue, G for Green and Y for Yellow. The color of the hidden blocks should be assigned in the form load procedure. The application interface should be clearly organized to maximize functionality.

5) Hidden Prizes/Concentration Application

Create a Hidden Prizes application that contains a control array of command buttons to represent a 5x5 board. The user is allowed 5 guesses to find the two randomly selected command buttons that contain the text "comp" and "uter". If the user finds both of the hidden words, he or she wins. The command buttons in the application should be greyed out once selected.

Enhance this application to be a game of concentration. Either decrease board size to 4x4 or have an open box. You can use pictures, numbers or words for matching purposes. Keep track of the number of attempts the user makes.

Optional Enrichment Projects are offered throughout the year as students finish assigned work early or for those students who wish to spend extra time in the computer lab during period 9. These projects allow students to further enhance their programming capabilities by using timers, random number generators as well as incorporating sounds and pictures into their programs. The students have the opportunity to write their own computer game software such as Tic-Tac-Toe (one and two player), Mouse/Space Invaders, Pong and Brick Breaker. More advanced games include Conway's Game of Life and Checkers.

Computer Programming 1
Pre-assessment

Name _____
Period _____

Place your answer to each multiple choice question in the space provided.

1)	2)	3)	4)	5)
6)	7)	8)	9)	10)
11)	12)	13)	14)	15)
16)	17)	18)	19)	20)
21)	22)	23)	24)	25)
26)	27)	28)	29)	30)

1. The word that indicates you are declaring a variable is:
 - a. Var
 - b. Variable
 - c. Dim
 - d. Declare
 - e. DecVar

2. Which of the following is not a valid data type:
 - a. Short
 - b. Long
 - c. Byte
 - d. Date
 - e. String

3. A variable that is declared without a stated data type is automatically given the data type:
 - a. Integer
 - b. String
 - c. Variant
 - d. Byte
 - e. Boolean

4. Which of the following cannot be done from the Menu Editor?
 - a. Create a menu
 - b. Create a submenu
 - c. Preview a menu
 - d. Add a shortcut key to a menu item
 - e. Add an access key to a menu item

5. Which of the following events are recognized by a menu item?
 - a. Click
 - b. Change
 - c. Enter key being pressed on the keyboard
 - d. both A and C
 - e. all three of A, B, and C

6. Visual Basic supplies a color constant for which of the following colors?
 - a. Orange
 - b. Cyan
 - c. Purple
 - d. Brown
 - e. Chartreuse

7. The key that is used most often in a shortcut, along with a letter is
 - a. Alt
 - b. Ctrl
 - c. Esc
 - d. F1
 - e. Shift

8. If a dialog box must be clicked before going on to the next page of a program, the box is
 - a. Enabled
 - b. Modal
 - c. Visible
 - d. Local
 - e. Global

9. The continuation character in Visual basic is the
 - a. Hyphen
 - b. Comma
 - c. Underscore
 - d. Dash
 - e. Number sign

10. The values returned by a MsgBox function of the OKCancel type are
 - a. 0 or 1
 - b. 1 or 2
 - c. 2 or 3
 - d. 0 or 2
 - e. 0 or 3

11. The InputBox has two buttons with captions
 - a. Yes and No
 - b. True and False
 - c. Cancel and Repeat
 - d. OK and Cancel
 - e. Ok and Retry

12. If a value is embedded in a string returned by the InputBox function
 - a. the Val() function will fail to convert it to a value unless the value is first in the string
 - b. the Val() function will fail to convert it to a value unless the value is last in the string
 - c. the Val() function will always fail to convert it to a value
 - d. it causes a run time error that crashes the program
 - e. the Val () function will return the currency value that follows a dollar sign

13. To open an existing sequential text file to read into the program use the Mode command:
- For Input
 - For Output
 - For Append
 - For Random Access
 - For Binary Access
14. The Set command
- Sets the value of a string
 - Assigns an object to an object variable
 - Creates a new text file
 - Reads the content of a text file
 - Sets a pathname to a particular file
15. When the Cancel button of the Open a File dialog box is clicked, the pathname returned is
- the null string with a length of 0 (zero)
 - the pathname of the most recently opened file
 - the pathname of the current project
 - unavailable to the programmer
 - that of the first available channel
16. The separate items in a Filter parameter string are separated with a
- Colon
 - Semicolon
 - Pipe symbol
 - Comma
 - Period
17. To connect a Visual Basic program with a database, the Data Form Wizard uses the
- ADO Data control
 - Standard Data control
 - Microsoft FlexGrid control
 - Grid control
 - Datalinked ListBox control
18. The statement MsgBox Err.Description displays the
- Error number
 - Location of the error handler
 - Description of the error provided by the Err object
 - Structure of the line that caused the error
 - Internal structure of the Err object

19. Me is a built-in object that represents the
- Current form
 - Currently open database
 - Data Form Wizard
 - Data form created by the Data Form Wizard
 - Jet database engine
20. To require prior declaration of variables names, include which statement?
- Require DIM statement
 - Option Explicit command at the beginning of each event procedure
 - Option Explicit command in the General Declaration section of the code
 - Dim statements at the top of the program code window
 - Dim statements for each variable at the beginning of every procedure
21. Errors in the structure of a statement are called
- Run-time errors
 - Value errors
 - Semantic errors
 - Syntax errors
 - Bugs
22. What does <F8> key do, when used in break mode
- Single steps through code line by line
 - Displays information about a selected variable
 - Skips over a procedure during line by line step through step of the code
 - Finds the next occurrence of a selected variable
 - Reports the values of all variables used in the line of code currently executing
23. Which of the following is a component of Object Oriented Programming?
- Data encapsulation
 - Inheritance
 - Polymorphism
 - All of the above
 - None of the above
24. Which of the following is not part of a class definition?
- Data
 - Events
 - Keys
 - Properties
 - Methods

25. The code **Dim objCar as New clsCar**
- Declares a new variable named **objCar**
 - Creates a new object according to a template provided by **clsCar**
 - Creates a variable of type **Car**
 - Is invalid syntax
 - Both a and b
26. The code **Dim insPolices(20) as clsInsurance**
- Creates an array of 20 objects of type **clsInsurance**
 - Creates an array of 21 objects of type **clsInsurance**
 - Creates an array capable of holding 20 object of type **clsInsurance**
 - Creates an array capable of holding 21 object of type **clsInsurance**
 - Is invalid syntax
27. Which of the following is considered a definite loop?
- If inttest=98 then MsgBox "You did a great job"
 - Do While intX>5
intX=intX-2
Loop
 - For intX = 1 to 10
Sum = Sum + intX
Next intX
 - Do
intX =intX-2
Loop While intX>5
 - Dim intX as Integer
28. How many times does the following loop execute?
For X = 0 to 7 Step 0.2
Y=Y+X
Next X
- 0
 - 7
 - 8
 - 35
 - 36
29. What is the value of 14 Mod 5?
- 2
 - 2.8
 - 3
 - 4
 - 9

30. Which of the following declares an array of integers named IntNumbers with 20 elements in it?

- a. Dim IntNumbers(19) as Integer
- b. Dim IntNumbers(20) as Integer
- c. Dim intNumbers(21) as Integer
- d. Dim intNumbers as 20 Integers
- e. Dim intNumbers as array of 20 Integers

Visual Basic 1
Midterm

Name _____

Part 1: Place your answers to part 1 on this answer sheet. 2 points each.

1.	2.	3.	4.	5.
6.	7.	8.	9.	10.
11.	12.	13.	14.	15.
16.	17.	18.	19.	20.
21.	22.	23.	24.	25.
26.	27.	28.	29.	30.

Part I: Answer all questions in this part. Place your final answers on the answer sheet.
Each correct answer will receive 2 points.

1. Indentation of code in Visual Basic
 - a. is necessary for this class.
 - b. is required by Visual Basic.
 - c. is done automatically by Visual Basic.
 - d. helps make code easier to read and understand.

2. The .bas file extension stands for
 - a. forms
 - b. Visual Basic project
 - c. Module
 - d. Data

3. To make a message box disappear, the user must
 - a. click the OK button
 - b. press the Escape key
 - c. press the Alt key
 - d. ask the teacher for help

4. A loop that depends on user input to end will often use a (n)
 - a. InputBox
 - b. MsgBox
 - c. Input statement
 - d. For-Next statement

5. If the condition tested in a Do While loop is true, the loop will
- not execute at all
 - execute at least one time
 - become an infinite loop
 - act like a For-Next loop
6. The menu item save would logically belong in which menu?
- File
 - Edit
 - Help
 - Window
7. The InputBox has two buttons with captions
- Yes and No
 - Cancel and Repeat
 - OK and Cancel
 - OK and Retry
8. Declaring a variable within a procedure of a code module ensures visibility
- throughout the code module
 - throughout the form
 - throughout all forms
 - only within that procedure
9. To give a variable maximum visibility within a form, you must declare it
- in each event procedure of the form
 - in the general declarations of the form with the Private keyword
 - in the general declarations of the form with the Dim keyword
 - in the general declarations of a code module with the Dim keyword

Visual Basic I Final Project

Your task is to write a user friendly, interactive program that meets the specific requirements described for that particular project. Each project will be graded out of 85 points. With each project there are optional extensions that can be done to add points to your grade. The maximum grade on any project is 100.

Once the program is completed you will hand in a printout of your code and form. Your program will be graded based on the following considerations.

- 1) Does your program run as specified in the description?
- 2) Is your code well thought out and organized?
- 3) Is the program user friendly?
- 4) Are the graphics appropriate?
- 5) Are any or all of the extensions completed?

Catch the Creature

Design and implement a game called Catch the Creature. Create a picture that represents a creature that can be displayed on the screen at any particular location. Have the creature appear at a random location on the screen for a random duration, then disappear and reappear somewhere else. The goal is to “catch” the creature by pressing the mouse button while the mouse pointer is on the creature image. Display an updated count for how many times the creature is caught. Have a new game button that resets the form.

Extensions:

- 1) Play an audio clip when the creature pops into another location.
- 2) Have more than one level.
- 3) Include a game over feature. Keep track of the number of creatures that a player misses. When the player misses 20 creatures (you can pick any reasonable value that you want) the game is over.

Rock Paper Scissors

Design and implement an application that plays the Rock-Paper-Scissors game against the computer. In the game, rock beats scissors, scissors beats paper and paper beats rock. The program should randomly choose one of the three options, without revealing it, and then prompt for the user's selection. At that point the program then reveals both choices and prints a statement indicating who won or if it was a tie. Continue playing until the player chooses to stop. Keep track and display the number of wins, losses and ties. Have a new game button that resets the form.

Extensions:

- 1) Create a two player option.
- 2) Display pictures of rock, paper or scissors at appropriate points during the game.
- 3) Try to create a strategy for the computer to use in the one player version.

Hangman

Write a program to play the game Hangman. Have one person enter the word to be guessed and then clear the screen. The other person then tries to guess the word by entering letters. Keep track of letters guessed in an appropriate display. As correct letters are guessed display the partial word on the screen. Only allow the user a limited number of guesses. Display a message of congratulations if the user guesses the word. Have a new game button that resets the form.

Extensions:

- 1) Have a visual representation of a hangman appearing one piece at a time corresponding to each incorrect guess.
- 2) Allow for a 1 player option. The computer will pick a random word from a pre made file of words.
- 3) Keep track of how many wins and loses in each session. Include an appropriate display

10. A given project can have
- a. only one form and one code module
 - b. as many form and code modules as wanted
 - c. a single form module and many code modules
 - d. a single code module and many form modules
11. Which of the following are not found in the Project Explorer window?
- a. View code button
 - b. View object button
 - c. A list of all forms and modules
 - d. A list of properties
12. In order to include a comment in your code, what symbol must you use?
- a. “
 - b. ‘
 - c. //
 - d. /
13. Command buttons let a user
- a. label a textbox
 - b. initiate program actions
 - c. fill a textbox with information
 - d. save a project
14. The property of a combo box that makes it most like a text box is:
- a. the AddItem method
 - b. the ListIndex property
 - c. the Top and Left properties
 - d. The Text property

15. To make a command button respond to a click when the program is running, the programmer must:

- a. provide the command button with an appropriate caption
- b. provide the command button with a different default
- c. provide the Click event procedure of the command button with code to execute
- d. provide the Change event procedure of the command button with code to execute

16. If the condition of an If statement is true, then

- a. the statements after **then** are executed
- b. the statements after **else** are executed
- c. it is matched with its corresponding **Case** statement
- d. no statements are executed

17. GUI stands for

- a. Graphics Unlimited Interface
- b. Graphical User Interface
- c. Graphics Underlining Interaction
- d. Graphical Underwriting Instrument

18. Which of the following cannot be done from the Menu Editor?

- a. add an access key to a menu item
- b. create a submenu
- c. preview a menu
- d. add a shortcut key to a menu item

19. An MDI can have how many forms associated with it?

- a. 0
- b. 1
- c. 2
- d. as many as you want

20. Which of the following is **not** found in the Visual Basic toolbox?

- a. ComboBox tool
- b. TextBox tool
- c. CreateBox tool
- d. ListBox tool

21. If a dialog box must be clicked before going on to the next part of the program, the box is

- a. enabled
- b. modal
- c. visible
- d. local

22. Arrays are used to

- a. store numbers
- b. store strings
- c. store forms
- d. both a and b

23. You refer to an element of an array by

- a. using the name of the array
- b. using the subscript of the element
- c. using both the name of the array and the subscript of the element
- d. its address in RAM

24. The scope of a variable is its:

- a. visibility and value
- b. visibility and lifetime
- c. lifetime and appearance
- d. lifetime and range

25. The InputBox function returns a value of what type?

- a. Integer
- b. Long
- c. String
- d. Single

26. How many times does the following loop execute?

```
For intX = 0 to 4 step 0.2
```

```
    IntY = intY + intX
```

```
Next intX
```

- a. 4
- b. 20
- c. 21
- d. 22

27. What is the value of x after the code fragment below finishes executing?

```
IntX = 5
```

```
Do
```

```
    IntX = intX - 2
```

```
Loop While intX > 5
```

- a. 0
- b. 1
- c. 2
- d. 3

28. The continuation character in Visual Basic is the

- a. hyphen
- b. comma
- c. underscore
- d. dash

29. Use a pop-up menu to:

- a. display any number of other pop-up menus
- b. display a menu of commonly used commands
- c. set the button images of a toolbar
- d. prompt the user to enter a string

30. The Len function is used to:

- a. count the buttons in the Buttons collection
- b. return the number of characters in a string
- c. count the files in the current directory
- d. concatenate strings

Visual Basic Midterm 2012 Part 2

Part II: Answer all questions on this part on the paper provided to you. Each problem requires you to write the code for a particular part of a program. Each question is worth 10 points. Partial credit will be given where applicable.

33. a) Write a For Loop that will add all even numbers from 2 to 200.

b) Write a Do While Loop that will add all numbers from 1 to 100.

34. To meet the minimum requirement of your first choice University, a student's SAT scores must add up to at least 1800. Write the code for the command button (**cmdtotal**) that reads the verbal, math and writing scores from three textboxes (**txtverbal**, **txtmath**, **txtwriting**) and tests the sum. If the sum meets the requirements, display a message of congratulations. If it does not, display a message of condolences.

35. Write the code for a button (**cmdrandom**) that would place 10 random numbers from 0 to 5 into an array (**myarray**). Use a For Loop. Then write a second loop that would place each of the numbers already stored in the array into a listbox (**myList**) next to its index. Put 2 spaces between the index and the value of the random number.

36. A person who shovels snow to make spending money wants to go to a concert. He figures that he will need \$175 for the tickets. Write the code for a button that repeatedly uses an InputBox to keep a running total of the money as he earns it, and counts the number of jobs. When the total amount exceeds \$175, display the amount he has earned and the number of jobs it took.