

A Study of the Programming Languages Used in Information Systems and in Computer Science Curricula

Jack Russell
jrussell@nsula.edu

Barbara Russell
brussell@nsula.edu

Lissa Pollacia
pollacia@nsula.edu

CIS, Northwestern State University
Natchitoches, LA 71497

William Tastle
tastle@ithaca.edu
Ithaca College
Ithaca, New York 14850

Abstract

This paper researches the computer languages taught in the first, second and third programming courses in Computer Information Systems (CIS), Management Information Systems (MIS or IS) curricula as well as in Computer Science (CS) and Information Technology (IT) curricula. Instructors teaching the first course in programming within a four year curriculum have wrestled with which language is best for enabling student success as well as preparing students for a viable career. The authors believe that Visual Basic.Net is the best language for teaching the first programming course. The authors believe that Graphical User Interface (GUI) controls within a Windows-based environment also should be emphasized in the first course. While the authors support JAVA and C# as advanced programming experiences within the CIS/MIS curriculum they do not believe that either of these two languages should be taught in the first programming experience. A survey was performed to determine: 1) Is there is a difference in the language used for the first programming course based on curriculum type (CIS, MIS/IS, CS and IT)? 2) Is there a difference in the language used for the second programming course based on curriculum type (CIS, MIS/IS, CS or IT)? 3) Is there is a difference in the language used for the third programming course based on curriculum type (CIS, MIS/IS, CS or IT)? The authors also want to know which interface (Windows/forms, Web or Console) is being used with the various programming languages being taught.

Keywords: computer languages, Visual Basic.Net, JAVA, C#, First Programming Language, Second Programming Language, Third Programming Language, Most Popular Language, Use Interface.

1. INTRODUCTION

Much has been written about the undergraduate curriculum in information systems. The IS 2002 Undergraduate Curriculum (Gorgone, Davis, Feinstein, Longenecker, 2002) details the IS Model courses and their content. The 2002 IS Undergraduate Model Curriculum IS 2002.5 course titled "Programming, Data, File and Object Structures" outlines key programming concepts and skills needed to be successful in information systems development. Appendix 1 describes this course.

The curriculum does not identify a preferred language for teaching this course. Having taught a series of programming languages in an undergraduate curriculum the authors recognize that the concepts within the IS 2002.5 course is often spread across a minimum of two learning experiences and sometimes across three courses. Many university curricula have a programming course in either the second semester of the freshman year or the first semester of the sophomore year. The second programming experience follows in the junior year with possibly a third course in the senior year.

1.1 The First Programming Course

The intent of the first programming experience is to introduce the student to basic Graphical User Interface concepts, simple programming concepts involving arithmetic/computations and basic comparison (decision making) that controls the execution flow of the instructions. This first course covers basic repetition concepts and structures and the use of both sequential files and simple array handling. This course may include basic object-oriented concepts that include 1) simple class creation and object instantiation and 2) the introduction to class libraries. The introduction of collection classes (Class Array, Class ArrayList) can be introduced as well.

1.2 The Second Programming Course

The second course continues to include more advanced array handling, sorting and searching techniques along with inheritance concepts using super classes and sub

classes. Procedure calls and parameter passing is normally covered. More advanced usage of class collections can be included such as hash tables, etc. This course often explores file handling in greater depth to include object instantiation, update and query.

1.3 The Third Programming Course

In CIS/IS curricula that aim to produce software engineers, a third programming course is often offered that covers ASP.Net and Web Services or through the use of the J2E platform or IBM's Websphere. Within the MS.Net framework, students learn to use Web Forms and Web Controls to create a web site. Students may also be exposed to Web Services for distributed processing across a network.

1.4 The key question: Which Language to Use for Advanced Programming?

Much has been written about which language should be used for each of the sequence of courses offered (Harkins, 2007; McKenzie, 2006; Zilbert, 2001). As far as the .Net Framework is concerned, all programming languages are equal on the basis of compilation and execution efficiency or their editing, design and debugging tools. Languages such as C#, Visual Basic.Net and others access the same classes and output the same MSIL (Microsoft Intermediate Language) code. The important differences between the languages are not to be found in their interaction with the runtime system but in the support they offer to the programmer or the student trying to learn the language. In short, the syntax, the structure and clarity of the language and ease of debugging the code becomes of paramount importance in choosing the proper language. The authors surveyed a number of post secondary institutions and found that the educator respondents strongly felt that Visual Basic.NET possessed the best language syntax and clarity of use for teaching students in the first programming course. Some of the reasons were that Visual Basic.NET was: 1) Not case sensitive (C# is case sensitive), 2) simple English-like statements, 3) simple loop structures with easy to remember begin loop names and end loop names. An easy illustration is shown comparing the syntax of

Visual Basic.Net and C# in Figure 1 and Figure 2. Figure 1 illustrates Visual Basic.Net syntax. Figure 2 shows the C# syntax to accomplish the same thing. Most readers can easily see the simplicity of the syntax of the Visual Basic.NET code as compared to the syntax of the C# code.

Figure 1: Visual Basic.Net Syntax

```

• Select Case color
  Case "pink", "red"
    r += 1
  Case "blue"
    b += 1
  Case "green"
    g += 1
  Case Else
    other += 1
End Select

```

Figure 2: C# Syntax

```

switch
(color) { //
  Must be integer or string
  case "pink":
  case "red": r++; break;
  case "blue": b++; break;
  case "green": g++; break;
  default: other++; break;
}

```

1.4.1 Interviews with IS Educators Teaching Programming Courses:

Six educators who teach in CIS/IS departments in colleges of business (Cook, et al., 2009) and have taught the beginning programming class unanimously agreed that the Visual Basic.NET syntax was much easier for students to learn over C#. Five of the educators had tried teaching either JAVA or C# in the beginning programming class and found significantly heavier drop rates and significantly lower test scores. Each of the five educators transitioned back to Visual Basic.NET as the language to use in the first programming class. Five prominent educators, from both CIS/IS and CS, felt that declining enrollments could be related to programs using C# or JAVA in the first programming class. Students become frustrated with the syntax and poor debugging tools associated with these languages, and as a result of this frustration, students often drop the course and switch majors. There were mixed feelings regarding

the second programming class with about the same number using Visual Basic.NET as those using C#. For those educators teaching a third programming class it was found that a small majority used either JAVA or C# along with ASP.NET. A large minority used Visual Basic.NET along with ASP.NET for distributed web projects.

1.4.2 Positive Comments about Visual Basic.NET from Educators

All of the educators interviewed agreed that Visual Basic.NET had the following positive features over C#:

1. The clarity of language syntax is less confusing to the student.
2. Students spent less time in the lab when using Visual Basic.Net.
3. Survey comments from students using Visual Basic.Net reported positive results.
4. Survey comments from students using C# reported less positive results.

1.4.3 IS Professionals Compare Visual Basic.Net and C#

Information Systems programmers (Microsoft Services, 2007) offer the following comparison of Visual Basic.Net and C#. These are listed to cast some of the reasons why educators have greater teaching success with Visual Basic.Net in the schools of business and schools of information sciences.

- 1) The Intellisense feature was the biggest advantage of Visual Basic.Net over C#.
- 2) Visual Basic.NET is NOT case sensitive. C# is case sensitive. Students or programmers must recognize that in C# a variable called *decSalary* as not recognized as the same variable as *decsalary* where they would be recognized as the same in Visual Basic.NET.
- 3) The default namespace is hidden.
- 4) Background compilation.
- 5) Simple syntax features over C#'s more tedious use of braces and brackets.
- 6) Visual Basic.Net terminates a block of code with the End Blockname statements (or Next statements, for a For loop) which are more familiar for programmers (or students) using SQL.

A few educators recognized certain features of Visual Basic.Net, but agreed that these reasons would not justify using C# in a beginning programming class. Some agreed that these features could warrant C# or JAVA being taught in an upper-level programming class. Some of the nice features of C# are:

1. Iterators and the Yield command.
2. Multi-line comments.
3. Implicitly Typed Arrays

2. SURVEY OF EDUCATORS TEACHING PROGRAMMING CLASSES

Educators were surveyed to determine the curriculum type (CIS, MIS/IS, CS, IT) and the computer languages they used in their first, second and third programming courses. From the survey 43% of the respondents were from CIS programs, 23% from MIS/IS programs, 13% from IT programs and only 3% from CS programs. Seventy-three percent (73%) of the respondents were from colleges/schools of business, and 90% were from 4-year undergraduate programs.

2.1 Survey Findings

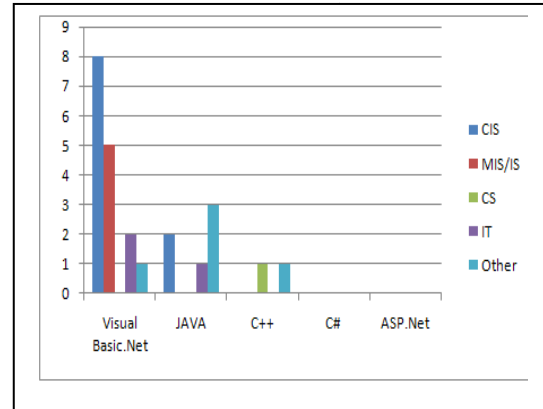
There were thirty (30) respondents from various universities and colleges. See Appendix 2 for the survey instrument employed for this study. Figure 3 shows the language that is most used in the first programming course in CIS is Visual Basic.Net (8) followed by JAVA (2). Five (5) respondents from MIS/IS report using Visual Basic.Net for their first course.

Figure 3: Number of Respondents Indicating which Programming Language used in First Course

	Visual Basic.Net	JAVA	C++	C#	ASP.Net
CIS	8	2	0	0	0
MIS/IS	5	0	0	0	0
CS	0	0	1	0	0
IT	2	1	0	0	0
Other	1	3	1	0	0

Figure 4 graphically measures the difference in the usage of the programming languages for the various programs.

Figure 4: Bar Chart of Language Usage between CIS/IS and CS



Within the CIS/MIS/IS programs, Figure 5 shows that there is a preponderance of teachers using Visual Basic.Net over JAVA with thirteen (13) using Visual Basic.Net and two (2) using JAVA.

Figure 5: Number of Respondents Indicating which Programming Language They Used in the First Programming Course within the CIS/MIS Curriculum

Language Used	CIS/MIS Respondents Using Language in First Course
Visual Basic.Net	13
JAVA	2
C++	0
C#	0
ASP.Net	0

Figure 6 graphically represents the number of respondents using Visual Basic.Net, JAVA, C++, C# and ASP.Net within CIS/MIS/IS.

Figure 7 shows the number of respondents indicating which programming language they use in the second programming course. Multiple languages could be selected for this response. We were interested in finding out in the second and third courses if educators were using multiple languages/platforms. Visual Basic.Net were used more in both CIS (10 respondents) and in MIS/IS (4 respondents), and with JAVA coming in second for CIS programs with four (4)

respondents. Across all programs Visual Basic.Net had a count of nineteen (19) while JAVA had a count of nine (9). ASP.Net came in third with six respondents. The survey allowed multiple languages to be chosen from for the second and third programming languages.

Figure 6: Bar Chart Depicting Respondents Use of Visual Basic.Net, JAVA and C# in the Classroom in CIS/MIS/IS Programs

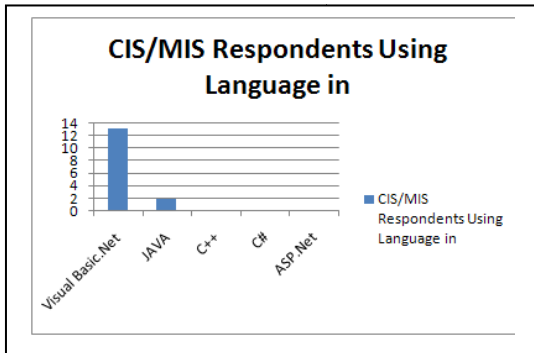


Figure 7: Number of Respondents Across Program Types Indicating which Programming Language they Used in Second Programming Course

	Visual Basic.Net	JAVA	C++	C3	ASP.Net	Total
CIS	10	4	1	3	1	19
MIS/IS	4	0	0	2	0	6
CS	0	1	0	0	0	1
IT	2	2	1	0	0	5
Other	3	2	1	1	0	7
Total	19	9	3	6	1	38

Figure 8 graphically illustrates the usage of programming languages in the second programming course. Visual Basic has a The survey did not attract many respondents from the computer science or IT programs; therefore, the tallies really don't tell us much. A second survey will be performed during the fall 2009 that will hopefully solicit more respondents from both CS and IT.

Figure 8: Bar Chart Illustrating the Usage of Programming Languages in the Second Programming Language for each Program Type

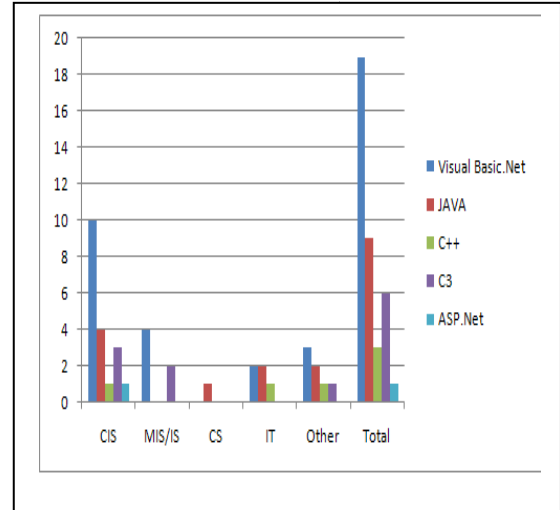


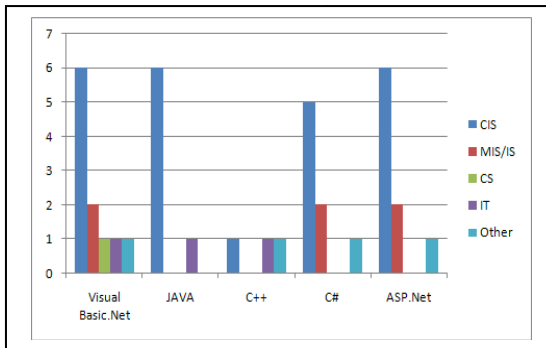
Figure 9 illustrates the number of respondents across the program types that indicate the programming languages used in the third programming course. The tally reveals that both Visual Basic.Net and JAVA are equally used as the primary language in the third programming course in CIS. MIS/IS programs still only used Visual Basic.Net for the third course.

Figure 9: Number of Respondents across Program Types Indicating which Programming Language they used in Third Programming Course

	Visual Basic.Net	JAVA	C++	C#	ASP.I
CIS	6	6	1	5	
MIS/IS	2	0	0	2	
CS	1	0	0	0	
IT	1	1	1	0	
Other	1	0	1	1	

Figure 10 provides a graphical comparison that shows a relatively equal usage of JAVA and ASP.Net in the third programming course. The client-side language Visual Basic.Net and the server-side language ASP.Net are extensively used. JAVA shows an equal usage in the third course in programming.

Figure 10: Bar Chart Illustrating which Programming Language is used in 3rd Course



The authors were also interested in knowing which user interfaces were used in conjunction with which language being taught for the first, second and third programming courses. Figure 11 shows that for Visual Basic.Net the Windows/Forms interface was the most popular while the Web and Console were favored by JAVA. Teachers teaching C++ favored the console interface. There were no respondents for C# or ASP.Net for this survey question.

Figure 11: Interface Used by Respondents Based on Language Used in First Programming Course?

	Visual Basic.Net	JAVA	C++	C#	ASP.Net
Windows/Forms	14	0	0	0	0
Web	1	1	0	0	0
Console	1	3	2	0	0
Other	0	2	0	0	0

The bar chart in Figure 12 illustrates the tallies shown in the table in Figure 11. For the first course the chart shows Visual Basic.Net and Windows/Forms as the favorite environment.

Teachers of programming often wrestle with which interface to use in a second programming course more than they do for the first programming experience. The authors wanted to know what interface was being used in the second programming experience. Figure 13 shows the tallies for the various interfaces being used for the various programming languages being taught. Windows forms and Visual Basic.Net seems to be the favorite environment for the

second programming course with fourteen respondents favoring this combination. There seemed to be little difference in the use of the web interface for the second course compared to the tallies from the first course (Figure 12).

Figure 12: Bar Chart of Which Interface Did Respondents Use for Which Language in First Programming Course?

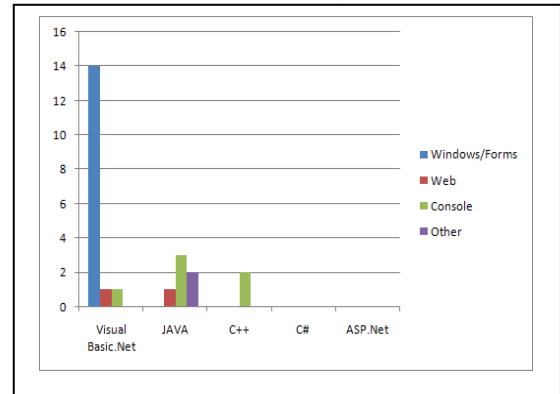
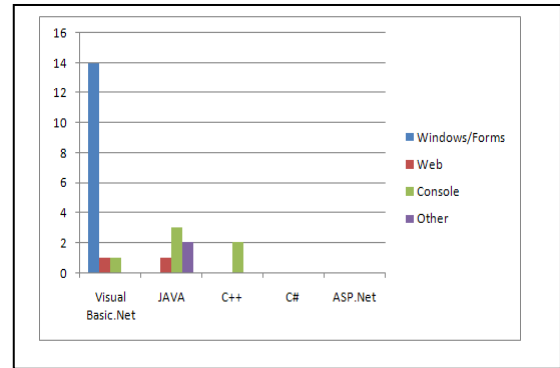
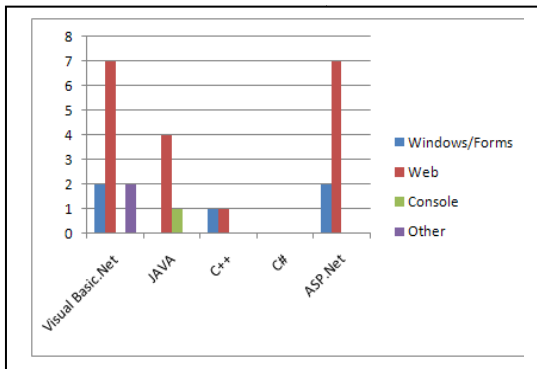


Figure 13: Interface Used by Respondents Based on Language Used in Second Programming Course?



As expected the web interface was preferred by most respondents who taught a third course in programming. Figure 14 illustrates the number of respondents using the various interfaces for the various programming languages being taught. The web interface was used two to three times as much as Windows/forms in the third programming experience.

Figure 14: Interface Used by Respondents Based on Language Used in Third Programming Course?



2.2 Hypotheses

Due to a small sample size (respondents = 30) and with certain expected frequency less than five (5) the authors chose to aggregate CIS and MIS and to also aggregate CS and IT. Based upon the general curriculum similarities between CIS and MIS and the similarities between IT and CS the authors chose to collapse these into two groups. Also, since most of the responses were for either Visual Basic.Net or JAVA, the authors chose to only look at the responses from these two languages for the first programming language. First, the authors were interested in knowing if there is a significant difference in the usage of Visual Basic.Net or JAVA in the first programming course based on the aggregated curriculum type [(CIS/MIS) or (CS/IT)]. The authors chose to limit the test to these two languages because visual inspection rules out the other languages. Several of the response tallies were less than two (2).

2.2.1 H₁ Null Hypothesis

Does program type influence the programming language used in the first programming course? The first null hypothesis states: "There is no significant difference ($p < = .05$) in the usage of Visual Basic.Net and JAVA in the first programming course based on curriculum aggregate type [(CIS/MIS or (CS/IT)]."

Testing the H₁ Null Hypothesis

The 2X2 table used to test the first hypothesis is shown in Figure 15. A Chi Square test of independence was done with this 2X2 table. A p value of .283 was calculated which does not allow the authors to reject the null hypothesis. The authors are unable to say with confidence that the program type influences the language used in the first programming course. Even with a Yates Correction of Continuity to adjust for expected frequencies less than five (5) there is no significant difference in the variables.

Figure 15: Respondents Using Visual Basic.Net or JAVA by Program Aggregate for First Programming Course

	Visual Basic.Net	JAVA	
CIS/MIS	8	2	10
CS/IT	5	0	5
	13	2	15
		p =	0.283

2.2.2 H₂ Null Hypothesis

The authors also want to know if program type influences the programming language chosen for the second programming course? The second null hypothesis states: "There is no significant difference ($p < = .05$) in the usage of Visual Basic.Net or JAVA in the second programming course based on curriculum aggregate type [(CIS/MIS or (CS/IT)]."

Testing the H₂ Null Hypothesis

The 2X2 table used to test the second hypothesis is shown in Figure 16. A Chi Square test of independence was done with this 2X2 table. A p value of .104 was calculated which does not allow the authors to reject the null hypothesis at the .05 level. The authors are unable to say with confidence that the program type influences the language used in the second programming course. Even with a Yates Correction of Continuity to adjust for expected frequencies less than five (5) there is no significant difference in the variables. It is worth noting that there is significance at

the .10 level that the program type influences the language used especially in the second programming course.

Figure 16: Respondents Using Visual Basic.Net or JAVA by Program Aggregate for Second Programming Course

	Visual Basic.Net	JAVA	
CIS/MIS	14	4	18
CS/IT	2	3	5
	16	7	23
		p =	0.104

2.2.3 H₃ Null Hypothesis

The authors also want to know if program type influences the programming language chosen for the third programming course? The third null hypothesis states: "There is no significant difference ($p < .05$) in the usage of Visual Basic.Net, JAVA and/or ASP.Net in the third programming course based on curriculum aggregate type [(CIS/MIS or (CS/IT))."

Testing the H₃ Null Hypothesis

The 2X3 table is used to test the third hypothesis (H₃). This is shown in Figure 17. A Chi Square test of independence was done with this 3X2 table. A p value of .346 was calculated which does not allow the authors to reject the null hypothesis at the .05 level. The authors are unable to say with confidence that the program type influences the language used in the third programming course. Even with a Yates Correction of Continuity to adjust for expected frequencies less than five (5) there is no significant difference in the variables. Moreover, simple visual inspection would reveal that Visual Basic.Net, JAVA and/or ASP.Net are almost equally used for the third course by CIS/MIS and also almost equally used in CS/IT for the third course. Since ASP.Net is a server-side language often used with Visual Basic.Net it is safe to say from the results that Visual Basic.Net is used for software development on the client-side while ASP.Net is used for software development on the server-side. It is worth repeating that the responses from the survey allowed multiple languages to be selected for the third programming course.

In other words, it is possible for a respondent to choose both Visual Basic.Net and ASP.Net for the third course in programming.

Figure 17: Respondents Using Visual Basic.Net, JAVA and/or ASP.Net by Program Aggregate for Third Programming Course

	Visual Basic.Net	JAVA	ASP.Net
CIS/MIS		8	6
CS/IT		2	1
		p =	0.346

3. ANALYSIS BY NOMINAL DISPERSION

When dealing with nominal scale data, such as in this study, it is expected that traditional statistical analyses be used in explaining the data. While statistical techniques do capture central tendency, variation, trend, etc, it is based on the existence of an interval or ratio scale. In the case of nominal data, there is no such interval and hence, there are really very few statistical measures that can be calculated on this kind of data. Mode is one such measure that can be calculated, but median and mean require the presence of an ordinal or interval scale, something not present in this data set.

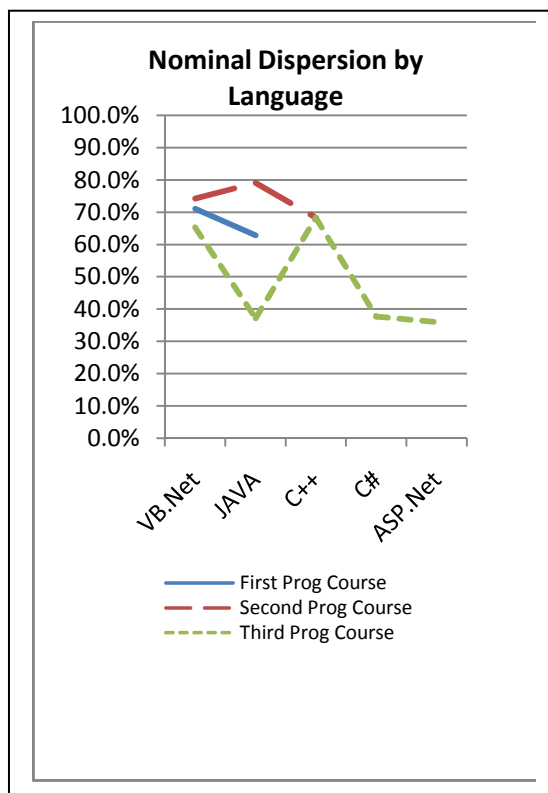
To indicate the degree of variance, a new measure is introduced here but only in the form of an illustration, for the mathematical proof and details are the subjects of another paper. Since we are dealing with nominal data this measure of variance is called *nominal dispersion* and is an application of the standard Shannon entropy equation, normalized by dividing the entropy by the log of the set cardinality (otherwise known as the maximum entropy). In short, the equation is

$$H(X) = -\sum_{i=1}^n p_i \log_2 p_i$$

H(X) is the usual representation for the Shannon entropy, p_i is the probability of the

i^{th} item in the distribution, and $|X|$ represents set cardinality.

This function, when normalized as shown above, produces a number in the $[0..1]$ interval. We then consider a value of zero to indicate an absence of dispersion (in statistical terms it is equivalent to a variance of zero), as occurs when all respondents have chosen the same category, or 1 when each respondent has chosen a separate category and each category has one and only one respondent. The results of a calculation of nominal dispersion on questions 1, 2 and 3 yield the following chart:



A quick review shows that Java, C#, and ASP.Net have the least dispersion. VB.Net has the most consistent nominal dispersion which is supported by the above analysis, and JAVA and C# have the least dispersion in the third course. Since this measure is currently under investigation, the authors are reluctant to make any further statements on the applicability of dispersion analysis on nominal data, but we open the door to future investigations involving nominal data.

4. CONCLUSION

For the first programming course in the CIS/MIS program area, the difference in the number of respondents for Visual Basic.Net and JAVA also seemed significant (8 vs. 2).

For the second programming course in the CIS/MIS program areas a visual inspection of the differences in the use of Visual Basic.Net and JAVA seems significant (14 vs. 4).

However, a Chi Square test of independence for the difference in proportions showed no significant difference when measuring frequency of CIS/MIS and CS/IT groups on the use of Visual Basic.Net or JAVA. This was especially true for the first programming course and the third programming course. Only the second programming course revealed a difference at the .10 level, but not enough to reject the null where the significance level was set at .05. So, only for the second programming course did program type seem to influence the programming language used.

The survey also revealed that the Windows environment was used more than other interfaces for the first and second programming courses while the Web interface was used much more often in the third level programming course.

5. REFERENCES

- Gorgone, Davis, Valacich, Topi, Feinstein and Longenecker (2002) 2002 IS Model Curriculum for Undergraduate Degree Programs in Information Systems, Copyright 2002, Association of Information Systems (AIS).
- Darrel Norton, "Visual Basic.Net vs C#: The Great .Net Language Debate", Darrel Norton's Blog, <http://codebetter.com/blogs/darrel.norton/archive/2003/07/07382.aspx>.
- Microsoft Services, "Differences Between Visual Basic.Net and C#", 2007, <http://support.microsoft.com/kb/308470>.
- Wikipedia, "Comparison of C Sharp and Visual Basic.Net, 2007,

- http://www.wikipedia.org/wiki/comparison_of_C_Sharp_and_Visual_Basic_.Net.
- Cook, Longenecker, Jeralds, Lee, Reithel, Sharp, Interview, April, 2009.
- McKenzie, W B. Introductory Programming with ALICE as a Gateway to the Computing Profession. In *The Proceedings of the Information Systems Education Conference 2006*, v 23 (Dallas): §2743. ISSN: 1542-7382. (A later version appears in [Information Systems Education Journal 7\(38\)](#). ISSN: 1545-679X.).
- Zilbert, A B. Integrating High Level Programming Languages For Teaching Business Application Development. In *The Proceedings of the Information Systems Education Conference 2001*, v 18 (Cincinnati): §22c.
- Harkins, R J. The Design and Implementation of a First Course in Computer Programming for Computing Majors, Non-Majors and Industry Professionals within a Liberal Education Framework. In *The Proceedings of the Information Systems Education Conference 2007*, v 24 (Pittsburgh): §2524. ISSN: 1542-7382. (A later version appears in [Information Systems Education Journal 6\(60\)](#). ISSN: 1545-679X.)

Appendix 1: The IS 2002.5 Software Development Course**IS 2002.5 – Programming, Data, File and Object Structures (Prerequisite: IS 2002.1)**

CATALOG: This course presents object oriented and procedural software engineering methodologies in data definition and measurement, abstract data type construction and use in developing screen editors, reports and other IS applications using data structures including indexed files.

SCOPE:

This course provides an exposure to algorithm development, programming, computer concepts, and the design and application of data and file structures. It includes the use of logical and physical structures for both programs and data.

TOPICS:

Data structures and representation: characters, records, and files; precision of data; information representation, organization, and storage; algorithm development; programming control structures; program correctness, verification, and validation; file structures and representation. Programming in traditional and visual development environments that incorporate event-driven, object-oriented design.

DISCUSSION: Students will gain in-depth understanding of defining and measuring events that produce data, both simple and complex, and principles, concepts, and practices of successful software development.

Formal problem solving strategies will be presented. Program design methods and strategies including top down implementation will be discussed and implemented. Graphic programming environments will be explored. Capabilities

of a number of programming languages will be presented. Skills will be developed in at least one language supporting an indexed file system.

Software engineering principles will be practiced with a systems view. Students will learn to recognize objects and abstract data types, concepts of event driven and data flow models, module identification, modularity including parameters, module naming, cohesion, coupling desired and erroneous practices, and testing. Verification and validation methods will be presented and practiced by generating small modules and larger programs.

Specific data structures including arrays, records, stacks, queues, and trees will be created and used. The course will provide an introduction to the use of predefined user interface components.

Appendix 2: Survey of Programming Languages Used in Computing Curricula**1. How is your program named?**

- .CIS (Computer Information Systems)
- .MIS/IS (Management Information Systems/Information Systems)
- .CS (Computer Science)
- .IT (Information Technology)
- .Other

2. Is your computing program within a College (or School) of Business?

- .Yes
- .No

3. Do you have a 4 year undergraduate degree program within your major?

- .Yes
- .No

4. What programming language is primarily used in your first programming course?

- Visual Basic.NET
- JAVA
- .C++
- .C#
- .Perl
- ASP.NET

.COBOL

Alice

.Other

5. Which environment is primarily used in teaching the first programming class?

.Windows/Forms

.Web Interface

.Console Interface

.Other

6. If applicable, which language is used in the second programming course? Multiple answers allowed.

.Visual Basic.NET

JAVA

.C++

.C#

.Perl

ASP.NET

.COBOL

Alice

.Other

7. If applicable, what environment is used in your second programming course?

.Windows/Forms

.Web Interface .Console Interface .Other

8. If applicable, what language is used in your third programming course? Multiple answers allowed.

 .Visual Basic.NET JAVA .C++ .C# .Perl ASP.NET .COBOL Alice .Other

9. If applicable, what environment is used in your third programming course?

 .Windows/Forms .Web Interface .Console Interface .Other

10. If you teach languages not specified in this survey, please list: