Programming for Humanists

Angelos Barmpoutis
Preeminence Associate Professor
Digital Worlds Institute / Online Learning Institute
University of Florida

Eleni Bozia
Assistant Professor
Department of Classics
University of Florida

```
1.  Dehinc Anno_Iuliano sit: 325 ↵
2.  Notula: First Council of Nicaea was in 325 AD
3.  Dehinc diebus sit: 0 ↵
4.
5.  Notula: Gregorian Calendar Adopted in 1582 AD
6.  dum ( Anno_Iuliano minus quam 1582 )
7.  incipit
8.      diebus augeatur modo: 0.0075 ↵
9.      Anno_Iuliano augeatur modo: 1 ↵
10. exit
11.
12. scribo ( cum stimulo: diebus ) ↵
```

How can we integrate computer literacy into Liberal Arts curricula?

How can we motivate students and scholars acquire coding skills?

What are the best practices for teaching computer programming to novices?

2/3 of the NEH funded Digital Humanities projects involve open-source computer code.

APIs (Application Programming Interfaces) have been developed for Database queries, Lexicographic analysis, GIS mapping, 3D structural analysis, etc.

- Open-Source APIs for the Advancement of the Humanities
- Is it accessible to the humanities scholars?

- Well studied
- Since ‘60s!
- Fein 1961
- Perlis 1962
- Licklider 1962
- DeBruijn et al. 1963
- Brandon 1962
- Buckingham 1965

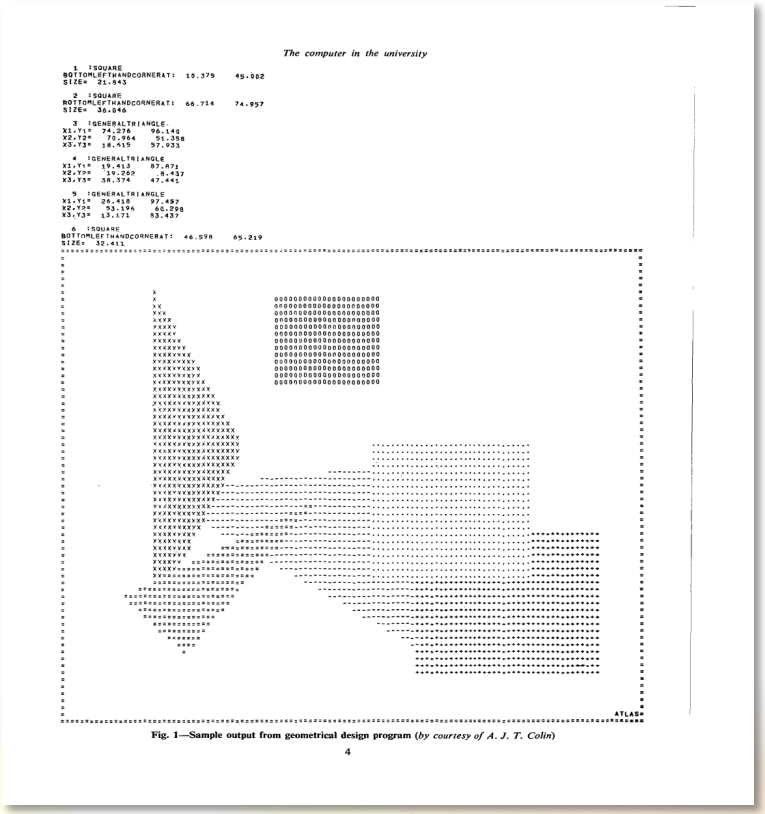


Fig. 1—Sample output from geometrical design program (*by courtesy of A. J. T. Colin*)
4

Alan Perlis argued that computer science should be part of a liberal education.

Explicitly, he argued that all students should learn to program.

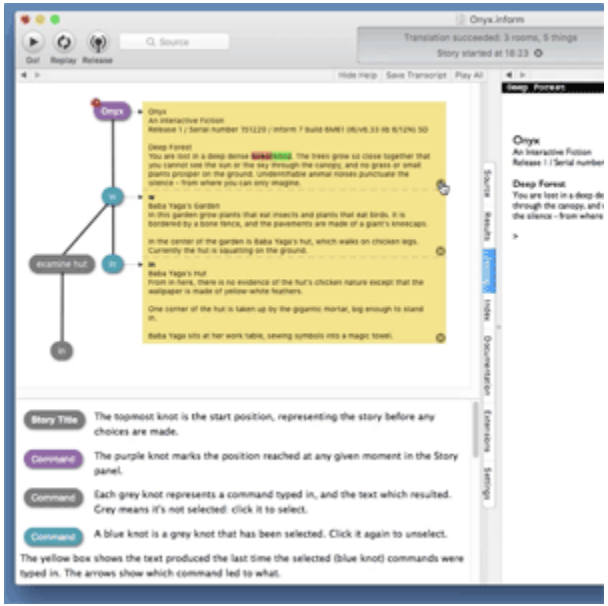
Early 1960's

- Today?
- Current topic in humanities forums.
- Discussions on “The Humanist”
  - Procedural Languages
  - Teaching Coding in Humanities

M. Meteas (2005) “all students, engineering and liberal arts students alike, should have a two semester computer science sequence in their freshman year."

- Inform7

- **Inform** is a design system for interactive fiction based on natural language.



- SPL

- “Shakespeare Programming Language” (SPL) is an esoteric programming language, which was designed to make programs appear to be similar to Shakespearean plays.

```
Romeo, a young man with a remarkable patience.
Juliet, a likewise young woman of remarkable grace.
Ophelia, a remarkable woman much in dispute with Hamlet
Hamlet, the flatterer of Andersen Insulting A/S.

                    Act I: Hamlet's insults and flattery
                    Scene I: The insulting of Romeo.
[Enter Hamlet and Romeo]
Hamlet:
You lying stupid fatherless big smelly half-witted cowa
stupid as the difference between a handsome rich brave
Speak your mind!
You are as brave as the sum of your fat little stuffed
old rotten codpiece and a beautiful fair warm peaceful
day. You are as healthy as the difference between the s
sweetest reddest rose and my father and yourself! Speak
You are as cowardly as the sum of yourself and the diff
between a big mighty proud kingdom and a horse. Speak y
Speak your mind!
[Exit Romeo]
                    Scene II: The praising of Juliet.
[Enter Juliet]
Hamlet:
Thou art as sweet as the sum of the sum of Romeo and hi
```

- Metafor
- Visualizing stories as code

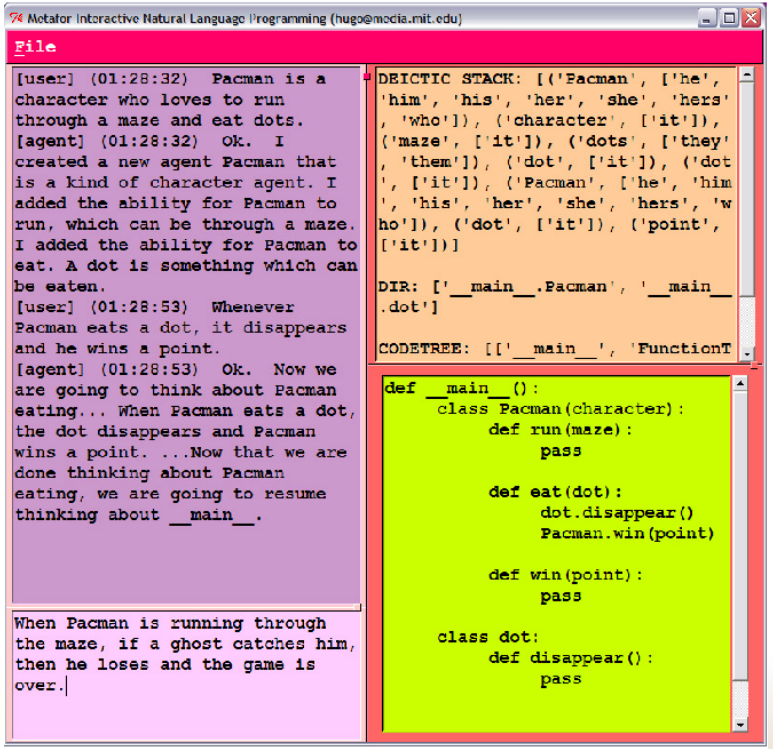


Figure 1. A screenshot of Metafor. Clockwise from the lower left

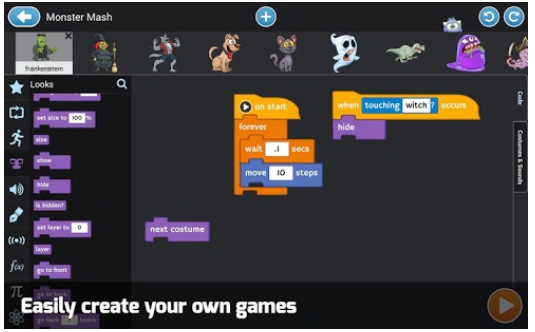
Can these tools motivate students and scholars of any discipline in the Humanities?

What about real programming languages? (Python, R, JavaScript, …)

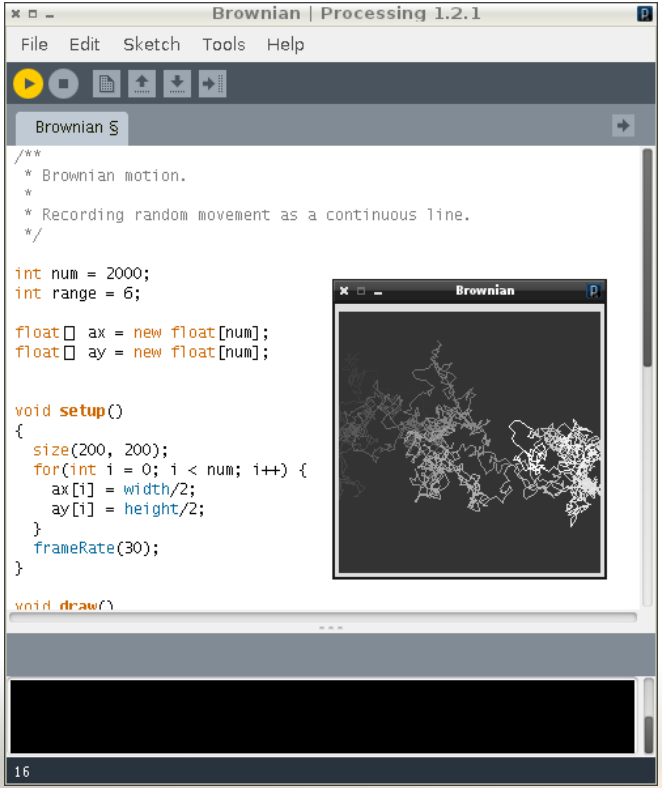
- Computer Education is a well studied topic
- Several approaches
- Main focus on K-12 education

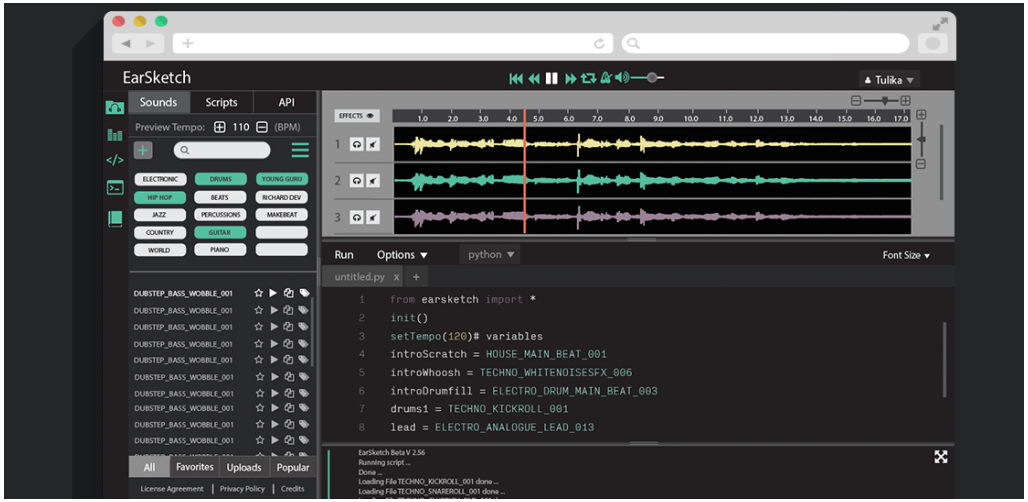
Tangible User Interfaces

Graphical User Interfaces



Text Editing Interfaces



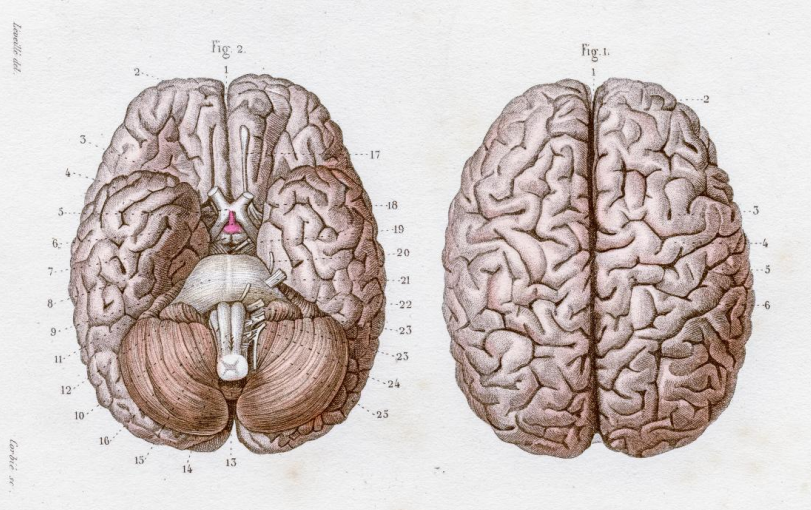


Coding vs Humanities?

- Need for coding in the Humanities?
- Best uses and best practices?
- How to motivate Humanities students?
- What is the natural creativity tool for the Humanities?
- Humanities vs Digital Humanities?

Theoretical Background

Reference: Angelos Barmpoutis at al., Human Factors and Ergonomics, 2017.




```
not:(or:|!be:)
```

not ☹ or 😐 ! be ☺

Theoretical Background

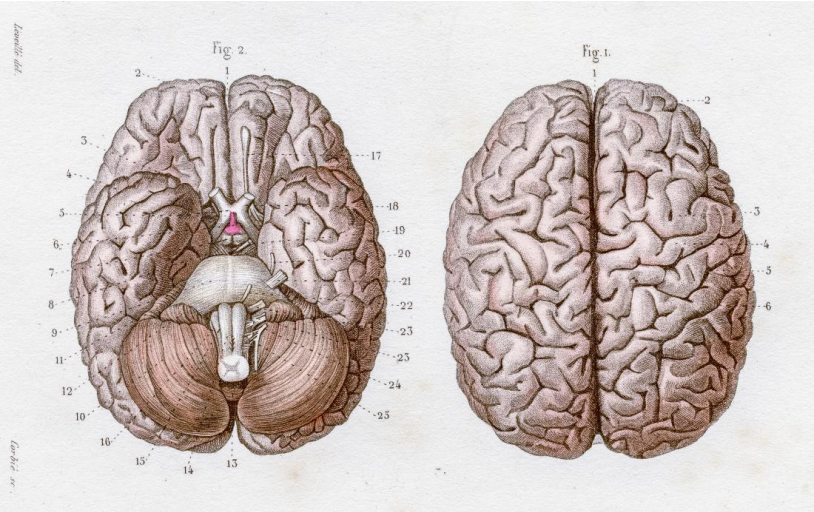
Reference: Kent Lyons, Thad Starner, Daniel Plaisted, James Fusia, Amanda Lyons, Aaron Drew, and EW Looney. SIGCHI 2004.

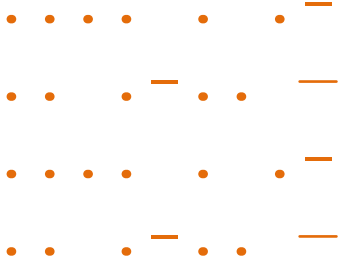
4433555
555666
4433555
555666

Hello
hello

Theoretical Background

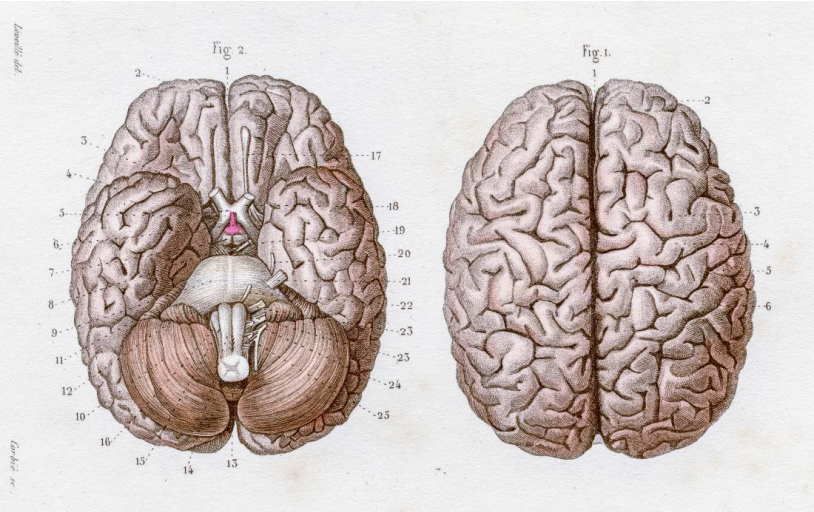
Reference: Lara Schlaffke, Alexander Leemans, Lauren M Schweizer, Sebastian Ocklenburg, and Tobias Schmidt-Wilcke. Frontiers in human neuroscience, 2017.





Theoretical Background

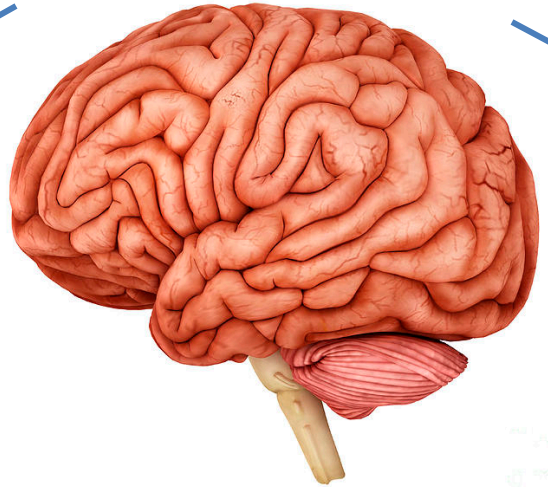
Reference: A. Stefik and S. Siebert. ACM Transactions on Computing Education, 2013.

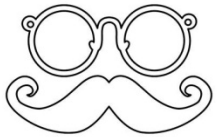



```
if        B8
else      7F
for       D0
while     62
int       CF
float     7A
```

Digital Worlds Institute | UF Classics University of Florida

Language Encoding





Generative skill

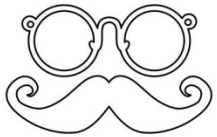
Comprehension skill

Ω

Symmetry

Identity mapping

Ω

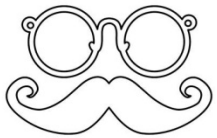


```
if a!=0
```

Symmetry

```
if a!=0
```

Identity mapping



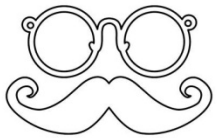
Theoretical Background

4433555555666

~~Symmetry~~

hello

some mapping



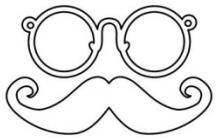
Theoretical Background

…. . .-.. .-.. —

~~Symmetry~~

`hello`

some mapping



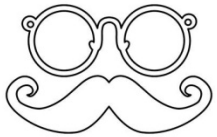
Theoretical Background

:(:| :)

~~Symmetry~~

☹ 😐 ☺

some mapping

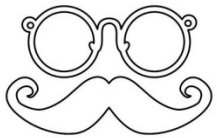


Theoretical Background

ALT+F4
COMMAND+W

~~Symmetry~~

some mapping



Proposed Method

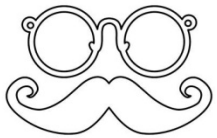
Proposed Method

```
if a!=0
```

~~Symmetry~~

if a is not 0

some mapping



Proposed Method

~~Symmetry~~

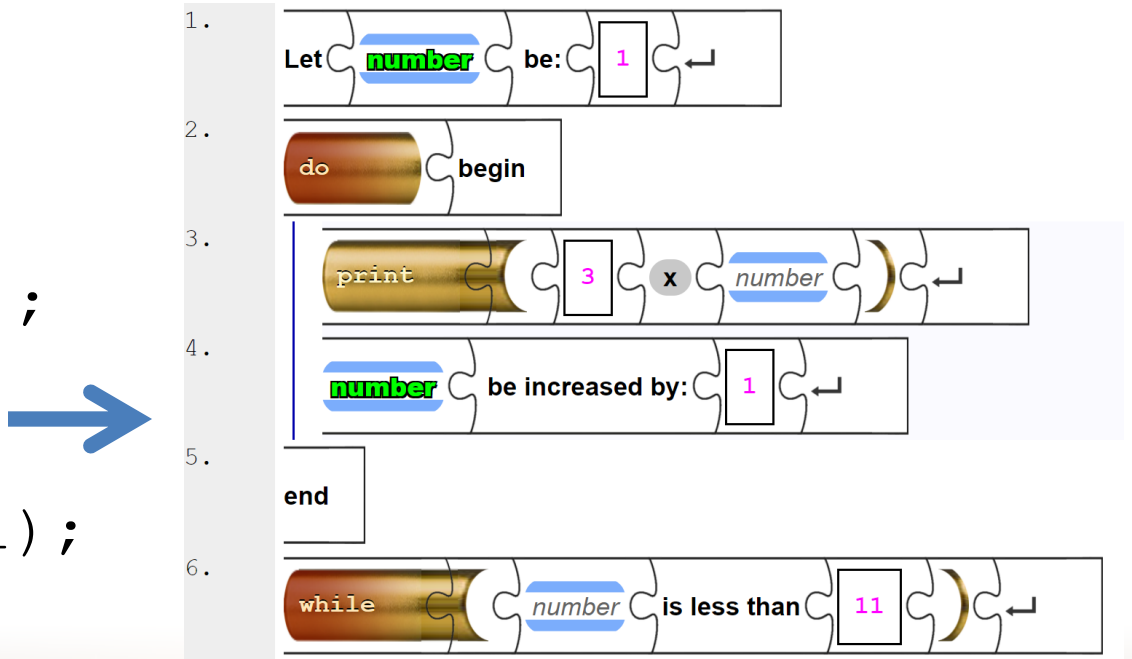
```
55      /**
56       * Code your drone to go to Koli and then back to Florida.
57       */
58      public void main(){
59          jump( speed: 5, degrees: 45);
60          flyTo( longitude: 29.8059383, latitude: 63.0955327);
61          rest( minutes: 90);
62          jump( speed: 5, degrees: 45);
63          flyTo( longitude: -82.3899576, latitude: 29.686252);
64      }
```

Symmetry

```
1   /**
2    * Code your drone to go to Koli and then back to Florida.
3    */
4   public void main(){
5       jump(5,45);
6       flyTo(29.8059383,63.0955327);
7       rest(90);
8       jump(5,45);
9       flyTo(-82.3899576,29.686252);
10  }
```

Proposed Method

```
1. var number=1;
2. do {
3. print(3*number);
4. number+=1;
5. }
6. while(number<11);
```



Proposed Method

```
var Julian_Year = 325 ;
var error = 0 ;

while ( Julian_Year < 1582 )
{
    error += 0.0075 ;
    Julian_Year += 1 ;
}

print ( error ) ;
```

Definiere: **Julian_Year** sei `325` ↵

Definiere: **error** sei `0` ↵

solange (*Julian_Year* ist kleiner als `1582`)

anfang

    error wird erhöht um `0.0075` ↵

    Julian_Year wird erhöht um `1` ↵

ende

print (*mit Parameter:* *error*) ↵

Proposed Method

Real JavaScript coding in asymmetric editor, a.k.a. Emoticoding

Proposed Method

Real JavaScript coding in asymmetric editor, a.k.a. Emoticoding

```
1.  Let bob be a new Robot ( ) ↵
2.  bob 's weight be 10 ↵
3.  bob do run ( ) ↵
4.
5.  while ( bob 's speed is not equal to 0 )
6.  begin
7.      bob 's weight be decreased by 1 ↵
8.      print ( bob 's speed ) ↵
9.  end
```

Proposed Method

Real JavaScript coding in asymmetric editor, a.k.a. Emoticoding

Proposed Method

Real JavaScript coding in asymmetric editor, a.k.a. Emoticoding

```
1.  Let bob be a new Robot ( ) ↵
2.  bob 's weight be 10 ↵
3.  bob do run ( ) ↵
4.
5.  while ( bob 's speed is not equal to 0 )
6.  begin
7.      bob 's weight be decreased by 1 ↵
8.      print ( bob 's speed ) ↵
9.  end
```

Table 4: Levels of student success per metric

Method	Metric 1	Metric 2	Metric 3	Metric 4
Nat. Language	52.38%	97.62%	80.95%	90.48%
Comp. Code	17.39%	93.48%	50.00%	54.35%
Increase	201.19%	4.43%	61.90%	66.48%
χ^2	11.9624	0.8675	9.2180	14.0898
p	0.00054	0.35164	0.00239	0.00017

of each group practiced using different text editors: the study group used the proposed editor with natural language replacements in English, and the control group used a conventional source code editor without replacements. Both editors used the same compiler (JavaScript compiler of Google Chrome) with the same error mes-

Fitting in Humanities Curricula

```
1.  Note: Romeo and Juliet by Shakespeare
2.  Let Act1 be the following procedure: ( )
3.  begin
4.      Let Sampson be: a new Character ( )
5.      Sampson 's props be: "Sword and Buckle"
6.
7.      Let Gregory be: a new Character ( )
8.      Gregory 's props be: "Sword"
9.
10.     Sampson do enter ( )
11.     Gregory do enter ( )
12.
13.     Sampson do say ( "Gregory, on my word, we'll not carry coals." )
14.     Gregory do say ( "No, for then we should be colliers." )
15. end
16.
17. play ( Act1 )
```

Fitting in Humanities Curricula

```
1. //Romeo and Juliet by Shakespeare
2. var Act1=function()
3. {
4.         var Sampson=new Character();
5.         Sampson.props="Sword and Buckle";
6.
7.         var Gregory=new Character();
8.         Gregory.props="Sword";
9.
10.        Sampson.enter();
11.        Gregory.enter();
12.
13.        Sampson.say("Gregory, on my word, we'll not carry
coals.");
14.        Gregory.say("No, for then we should be colliers.");
15. }
16.
17. play(Act1);
```

Fitting in Humanities Curricula

```
1.  Dehinc Anno_Iuliano sit: 325 ↵
2.  Notula: First Council of Nicaea was in 325 AD
3.  Dehinc diebus sit: 0 ↵
4.
5.  Notula: Gregorian Calendar Adopted in 1582 AD
6.  dum ( Anno_Iuliano minus quam 1582 )
7.  incipit
8.      diebus augeatur modo: 0.0075 ↵
9.      Anno_Iuliano augeatur modo: 1 ↵
10. exit
11.
12. scribo ( cum stimulo: diebus ) ↵
```

Fitting in Humanities Curricula

```
1.  var Anno_Iuliano=325;
2.  //First Council of Nicaea was in 325 AD
3.  var diebus=0;
4.
5.  //Gregorian Calendar Adopted in 1582 AD
6.  while(Anno_Iuliano<1582)
7.  {
8.   diebus+=0.0075;
9.   Anno_Iuliano+=1;
10. }
11.
12. scribo(diebus);
```

Fitting in Humanities Curricula

- Use natural languages as the bridge between Liberal Arts and Coding

- Disciplines that study languages
- Disciplines that use languages as tool
- Disciplines that use language for creative writing and critical thinking

- Natural Languages is the common element.

Fitting in Humanities Curricula

JavaScript

```
1.  var number = 0 ;
2.  while ( number < 10 )
3.  {
4.      number += 1 ;
5.  }
```

English

```
1.  Let number be: 0 ↵
2.  while ( number is less than 10 )
3.  begin
4.      number be increased by: 1 ↵
5.  end
```

Latin

```
Dehinc number sit: 0 ↵
dum ( number minus quam 10 )
incipit
    number augeatur modo: 1 ↵
exit
```

Ancient Greek

```
Ὁρίσωμεν ὅτι το number ᾖ: 0 ↵
ὅσον ( το number ἐλάττον τοῦ 10 )
ἀρχή
    το number αὐξάνηται κατά: 1 ↵
τέλος
```

- Is straight alignment possible?
- Grammar or Syntax problems?
- Noun declension?
- What natural languages can be used?
- What programming languages can be used?

Thank you

Programming for Humanists

Angelos Barmpoutis
Preeminence Associate Professor
Digital Worlds Institute / Online Learning Institute
University of Florida

Eleni Bozia
Assistant Professor
Department of Classics
University of Florida