

Logicus

Logicus 2.0 is an educational software for beginners in programming world, intending to simulate any logical scheme. To view the source code in AILab, Pascal, C / C + +, pseudocode for logical scheme constructed by the student with drag&drop.

What is an educational software 2.0?

Software, which contains the paradigm of artificial intelligence and allow the student to study the notions elements created by him. Not based on content, (for information is Google and other search engines), the software allows maximum interactivity and can be created a thousand of examples. The software is not limited to a few animations, or frames.

The screenshot displays the Logicus 2.0 software interface. The main window shows a flowchart for a loop algorithm. The flowchart starts with a 'Start' terminal, followed by an assignment 'n=4', then 'i=0'. A loop structure is formed by a connector, an assignment 'i=i+1', a 'Write i,\n' output, and a decision diamond '(i<=n)'. The 'Yes' path loops back to the connector, and the 'No' path leads to a 'Write "am iesit",\n' output and an 'Exit' terminal. The interface includes several panels: 'Object Inspector' on the left, 'Source in Pascal' on the right showing the corresponding Pascal code, 'Global options' at the bottom right, and a 'View VariablesForm' table at the bottom center. A simulation window on the left shows the output of the program.

```
graph TD
    Start([Start]) --> N4[n=4]
    N4 --> I0[i=0]
    I0 --> Connector(( ))
    Connector --> Iplus[i=i+1]
    Iplus --> Write1[/Write  
i,\n/]
    Write1 --> Decision{i <= n}
    Decision -- Yes --> Connector
    Decision -- No --> Write2[/Write  
"am iesit",\n/]
    Write2 --> Exit([Exit])
```

```
Source in Pascal
Source
Program LogicusSchemeConverted;
var
  n, i: Integer;
Begin
  n:=4;
  i:=0;
  i:=i+1;
  WriteLn(i);
  While (i<=n) do
  Begin
    i:=i+1;
    WriteLn(i);
  End;
  WriteLn('am iesit');
  Exit;
End.
```

Name	Variable type	Array dimensions	Value
n	Integer		(0)
i	Integer		(0)

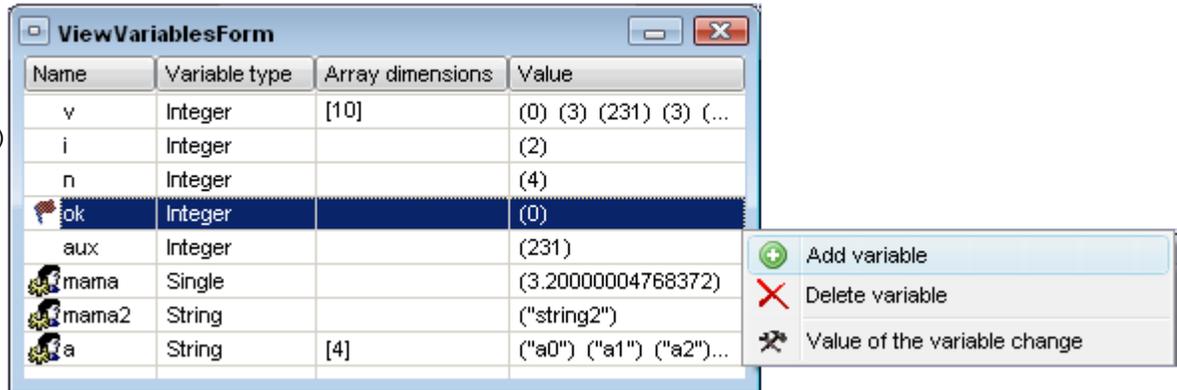
Simulation of scheme logic !
[Hint] Start from Start1
am iesit
[Hint] Exit from Exit1

Name	Variable type	Array dimensions	Value
n	Integer		(0)
i	Integer		(0)

Variable table

To add variable follow: Watch → PopMenu → Add variable. Software currently allows eight types of data, each data type can be one-dimensional arrays, two dimensional, three-dimensional

Integer
SmallInt
ShortInt
Int64
Byte
Word
LongWord
Real(Single)
String



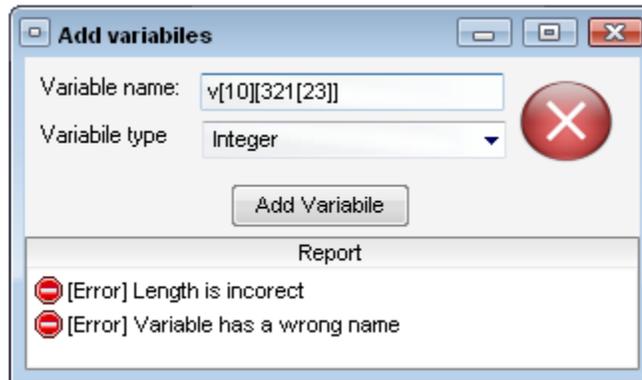
The 'ViewVariablesForm' window displays a table with the following data:

Name	Variable type	Array dimensions	Value
v	Integer	[10]	(0) (3) (231) (3) (...)
i	Integer		(2)
n	Integer		(4)
ok	Integer		(0)
aux	Integer		(231)
mama	Single		(3.20000004768372)
mama2	String		("string2")
a	String	[4]	("a0") ("a1") ("a2")...

A context menu is open over the 'ok' variable, showing the following options:

- Add variable
- Delete variable
- Value of the variable change

(Watch)



The 'Add variables' dialog box shows the following input fields:

- Variable name: v[10][321[23]]
- Variable type: Integer

The 'Add Variable' button is disabled. A red 'X' icon is visible in the top right corner. Below the input fields, a 'Report' section displays the following error messages:

- [Error] Length is incorrect
- [Error] Variable has a wrong name

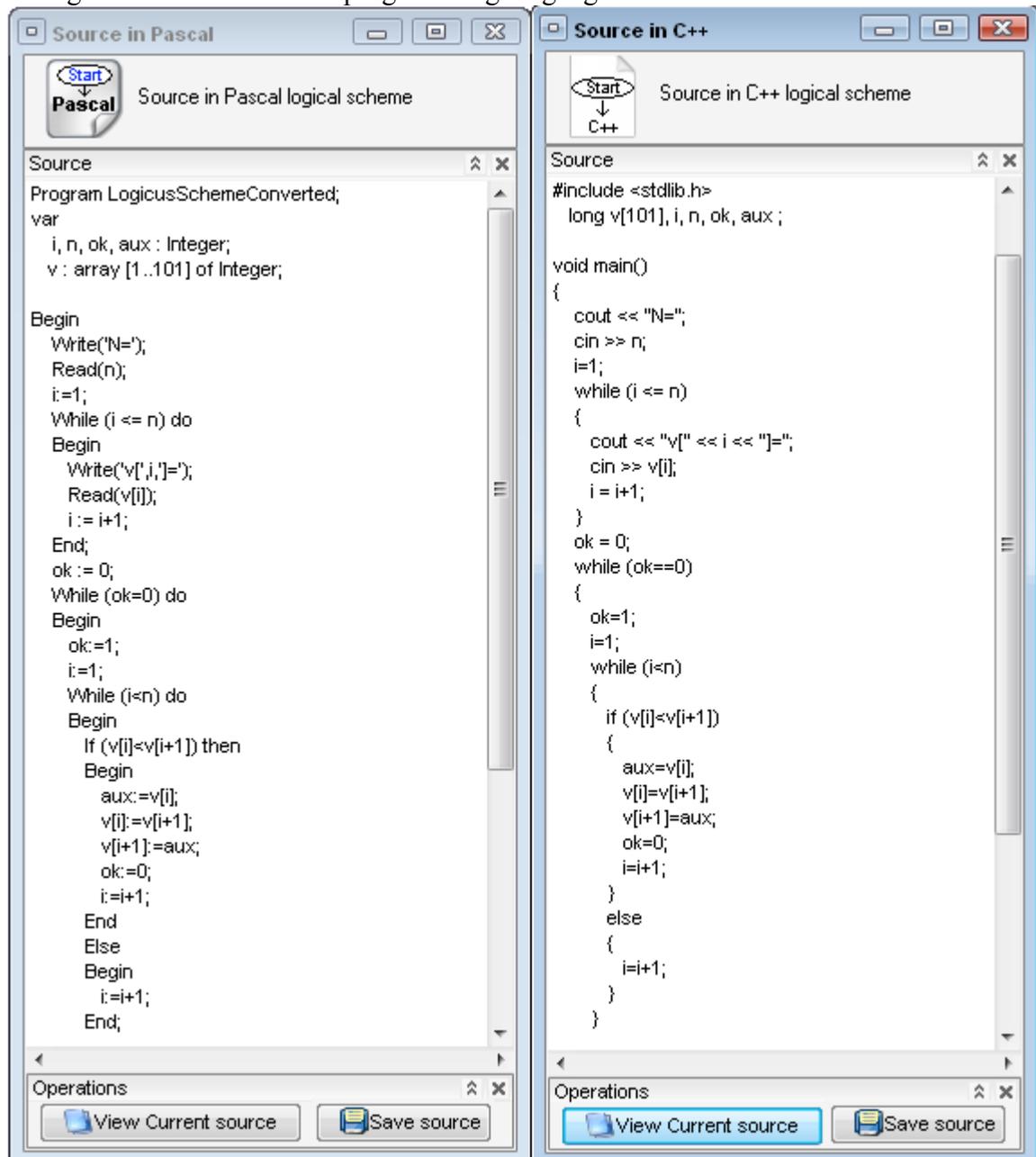
When introducing a new variable will be verified, if not exist, if lexic/syntax is valid, and the size. In this image, software shows that the variable can not be declared because the variable name is used, and dimensions are not good (parentheses are not closed properly).

Logical scheme simulation

Application player. Simulate the created scheme by user. The simulator is a synchronized thread. In simulator can set time to wait after execute next instruction. During the execution variables can be set manually from the watch.

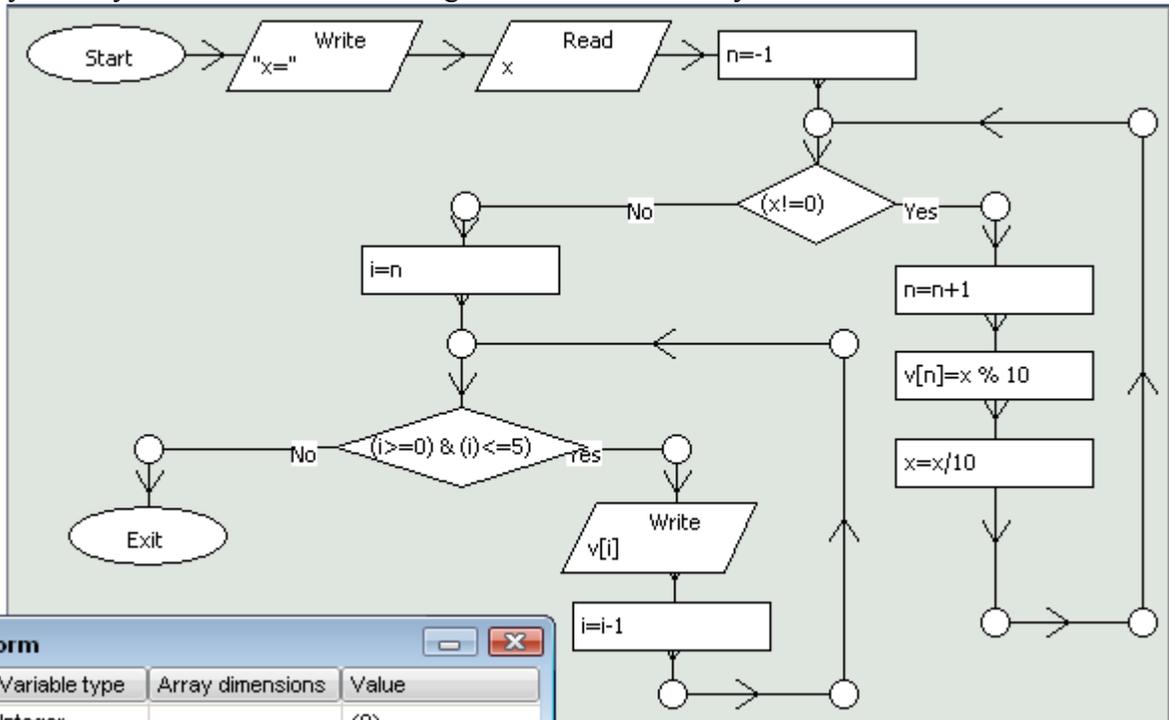


Export the logical scheme in various programming languages.



Lexical, syntactic analyzer.

The software has a synchronized thread, which all the time concurrent make the analysis of syntax and lexical of the logical scheme created by the student.



Name	Variable type	Array dimensions	Value
n	Integer		(0)
i	Integer		(0)
v	Integer	[10]	(0) (0) (0) (0) ...

Software automatically, detects these errors in user created scheme.

Messages

- Undeclared variable x at object Read1
- Undeclared variable x at object If1
- Undeclared variable x at object Attribute3
- Undeclared variable x at object Attribute4
- Expression is invalid, brackets are incorrect at object If2

Object selected: Read1

Artificial Intelligence.

The software contains a single paradigm of artificial intelligence, namely the perceptron. Neural networks were created Multi-Layer Perceptron in NeuroLab were trained in NeuroLab (application written by me and presented at many competitions), and then exported. I wrote the classes which allow to read the network weights trained, and to simulate:

I used in:

Speech recognition, allowing the recognition of words:

- Search
- Delete
- Save
- Load
- New
- Stop

Optical Character Recognition, which allows the student to draw numbers and the software can recognize.

Face recognition can recognize human faces.

Vocal User interface

VUI is a new concept of interface, which allows voice commands and interface for generate audio responses for user. To generate sound of the words I wrote my own narrator who speaks only in Romanian, it speaks with my voice, I don't use neural network, I use my own idea to write. Unfortunately, I don't have an interpretation of human language, interface recognize only some words. Facial recognition software allow me to create a face classification based on logging from faces. I wanted to show that an area so close, educational software can have innovations. I showing a new direction for the development of educational software.

Speech recognition

As ANN can only classify, so I have to turn my problem into a classification problem. Sound must be turned into a picture. Neural network will try to classify the picture. Creating a picture in amplitude-time domain is not useful, i need to work in the frequency-time domain.

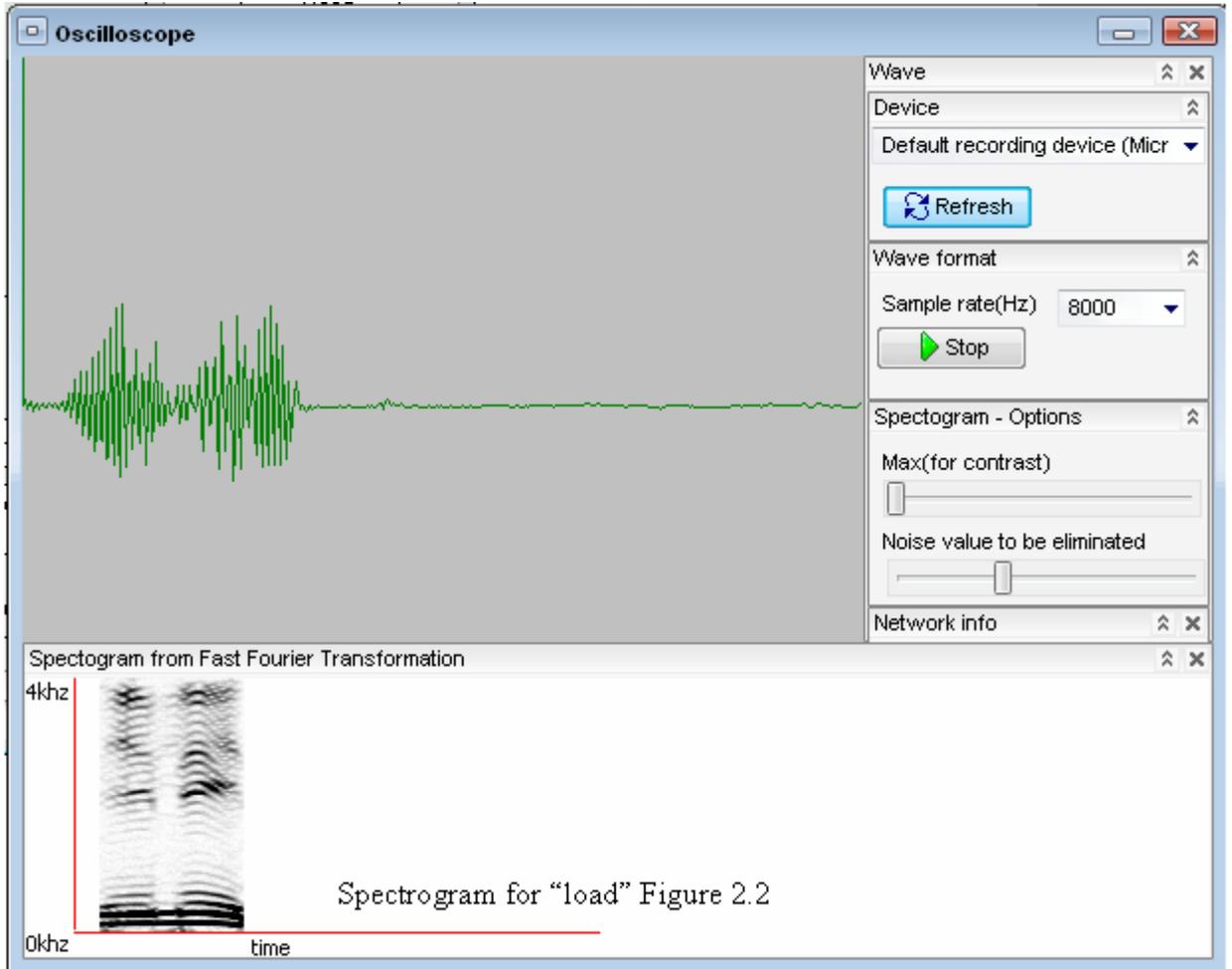
After that I wanted to see if a MLP neural network can classify sounds. So I created an Oscilloscope in time domain(OX) and amplitude(OY). I worked with Microsoft Multimedia System (mmsystem) taking samples from the microphone. Samples are on 8 bits. I set the sampling frequency to 8000 Hz. Then I created a class that draws samples. When I drew samples I did not remove the down component (DC) [Figure 3.1]. Then I deleted DC. I multiplied with Hamming Window to improve quality of recorded sound from the microphone..

Hamming window is: $\text{hamming}(x)=0,54-0,46*\cos(2*\pi*(x/256))$ [Figure 3.2]. This function is classic. I read about it from Matlab.

Algorithm with Hamming window: Take the first 256[1..256] samples inserted them into a buffer and multiply with Hamming values. After calculating the first window. I calculating Fast Fourier Transformation for first window. From the results obtained I will calculate the module from complex numbers. Obtaining one column from the gray-scale image(sound spectrogram). So I will get 8000(number of samples taken in one

second) / 30, (where 30 is the displacement of Hamming window) [Figure 3.1]. Then I repeated the process at next interval [30,286] and applied the same algorithm. Then [60,316], etc. ...

Values we obtained in spectrogram, it will be normed, and multiply by 255 to transform them into values from [0..255]. I applied different methods to remove noise with mediation, down component (DC), and separate words from break areas. I applied various levels of separation of noise and break interval. I used the first and the last component from the entire application, for FFT. In the future I will try to write it myself, so the FFT code is not write by me, is implemented by Dan Cross



After getting the spectrogram, and separated in the words, I will resize the spectrogram as image to an standard dimension(because, the input layer is formatted from 1218 inputs), to 42*29 pixels, because for computer algorithm the information is redundancy so information from the first spectrogram, and the second spectrogram is the same.



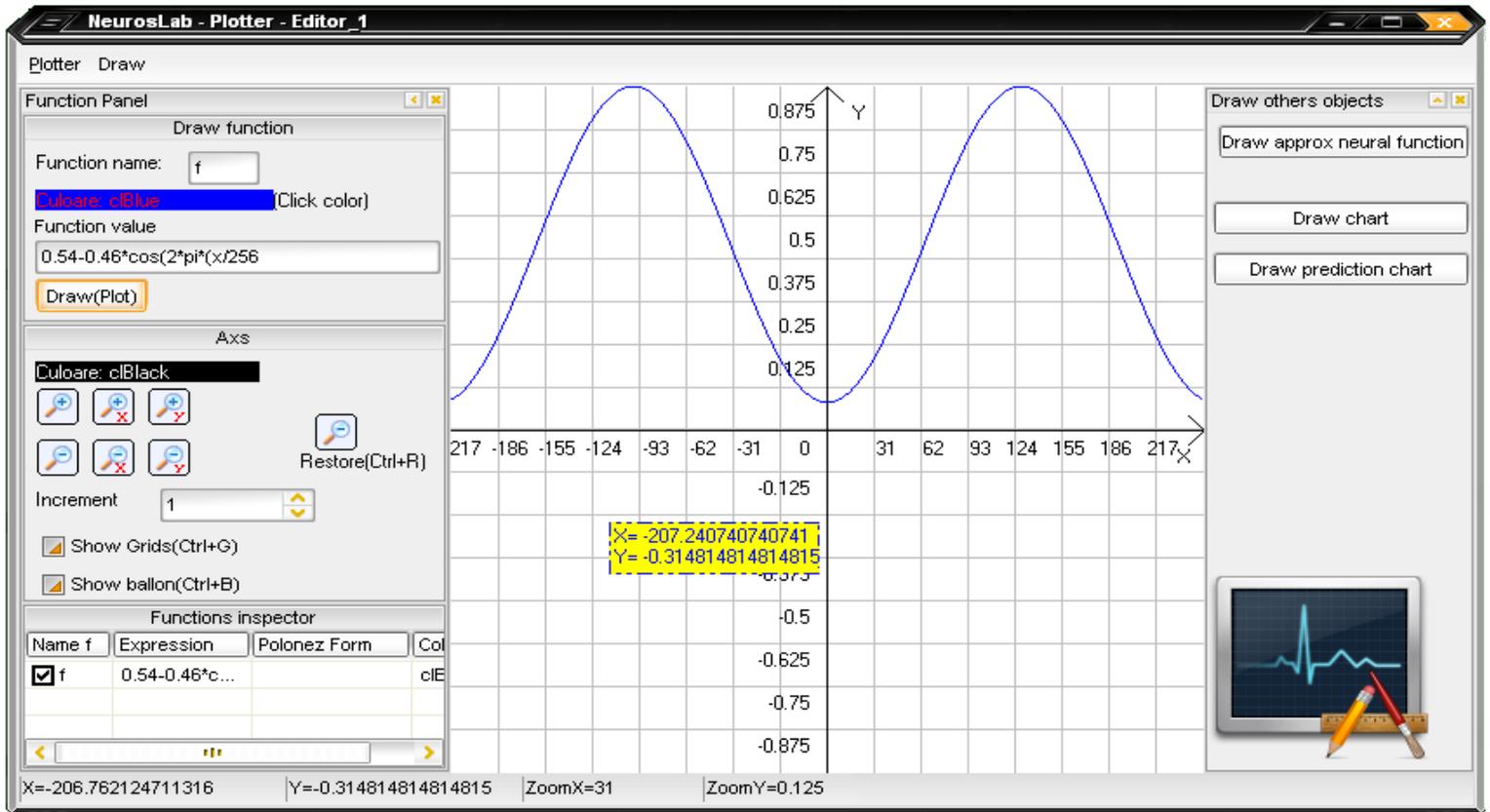


Figure 3.1 hamming plotted in NeurosLab(my software)

Educational part:

The software generates a type of educational test that generate a random logic scheme and a student must reposition and is obliged to respond by executing code what value have a variable. The student completes by handwritten the result, using an OCR (written by me) software can recognize the input, and the input is spoken by a narrator(written by me) in Romanian language.

The OCR have 100 inputs, and 10 outputs(because there are 10 classes[number from 0 to 9]). Student will draw the answer in a canvas form(dimensions 128*128 pixels). After the student draw, I make a Regional of interest(ROI) I will crop the number, rescaling the image to a small image(10*10pixels) because for computer algorithm the information is redundancy so information from the first image, and the second image is the same.

On the training the network was learned with 5 examples for each class, so 50 examples.

Da3

Software Help

Culoarea fundal Culoare text Culoarea linie Show variables

Main Menu

- Teste generate random
 - Test determinare variabile
- Teste încărcate
 - Test cu coloane 1 - Example
 - Test cu coloane 2 - Scheme log
 - Lectie educativa
 - Lectie educativa noua

Test determinare variabile 1

Problema

Question: Completeaza in urmatoarul edit, valoarea variabilei portocala

portocala =

0 4 5

VerificaButton

Save

Clear button

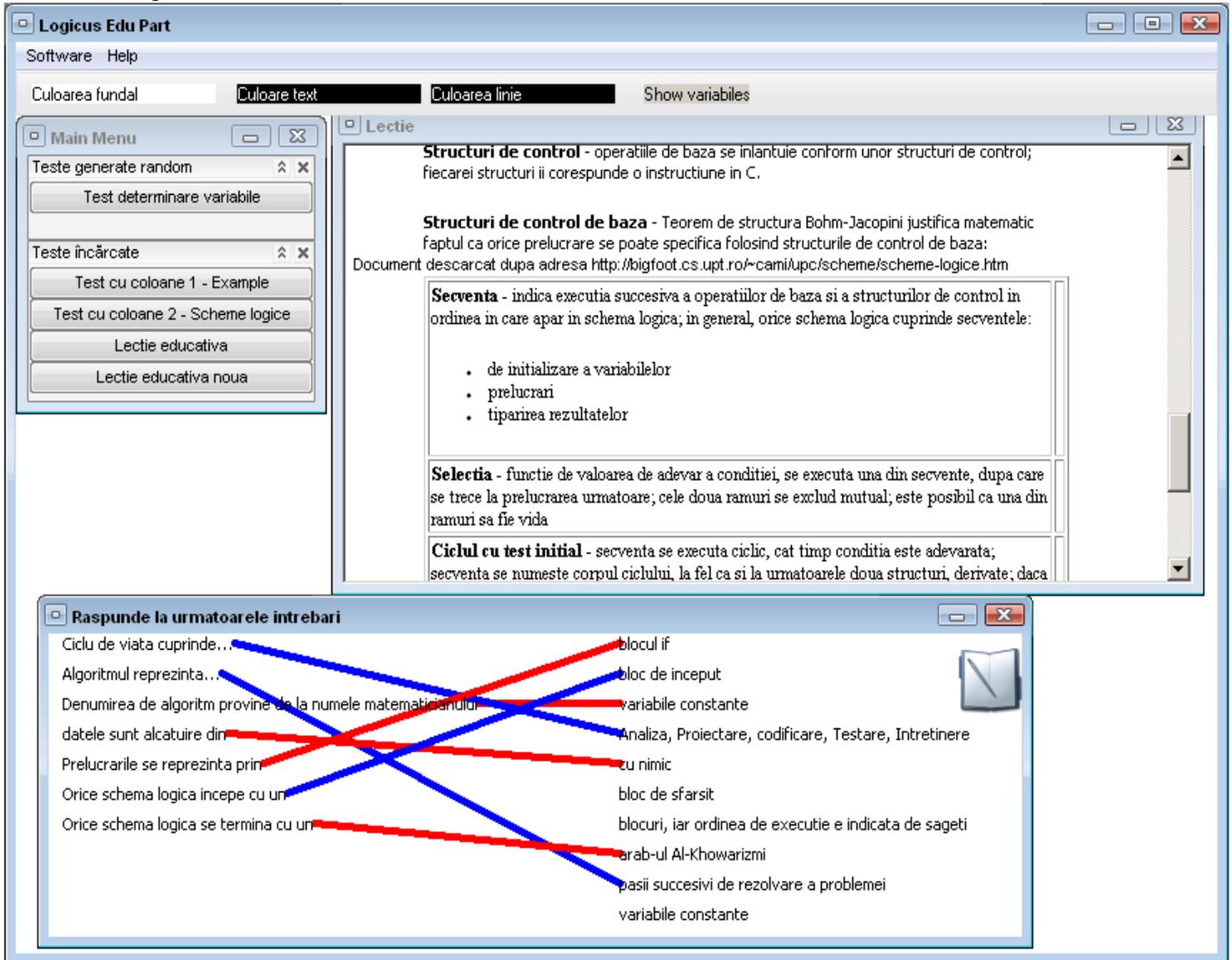
0 4 5

```

graph TD
    Start([Start]) --> Test["test=12-9"]
    Test --> Portocala1["portocala=11+test-3+10+3"]
    Portocala1 --> Portocala2["portocala=11+portocala+4+6"]
    Portocala2 --> Banana["banana=11+8"]
    Banana --> Tata["Tata=9+banana+test+test"]
    Tata --> Exit([Exit])
  
```

Object selected: Attribute2

The teacher can create tests (with answers column) and a document lessons (files supported by IE browser, so flv, html, pdf, doc). The teacher writes a XML file and the software parse and create lesson. Lessons can also download from update form . Logicus represents a new vision of educational software.



```
<Test with colons 1 - Example>
<Test Info>Test with Colons</Test info>

  <Colon A>
    <Element 0>
      <Caption>Question 1</Caption>
      <Answer>Answer 1</Answer>
    </Element 0>

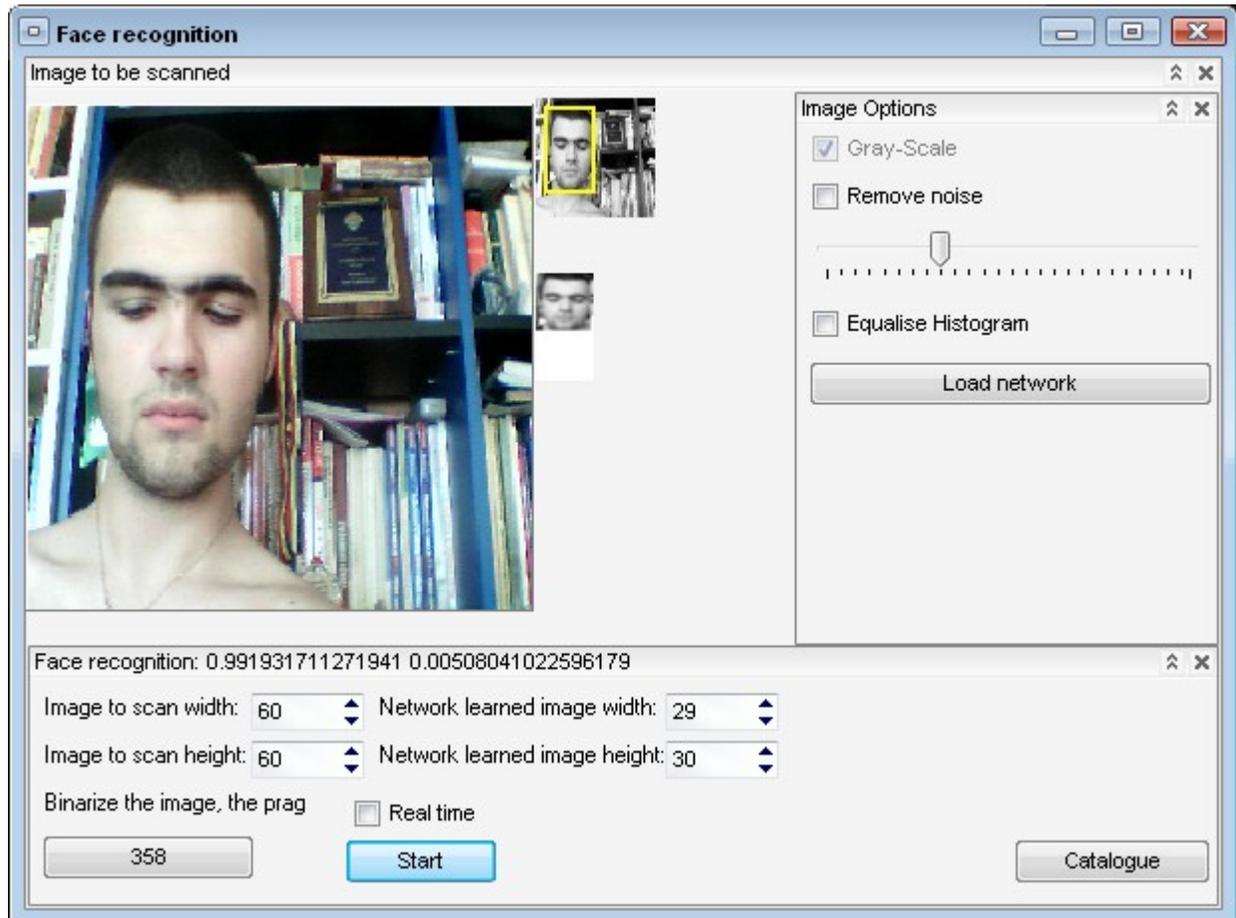
    <Element 1>
      <Caption>Question 2</Caption>
      <Answer>Answer 2</Answer>
    </Element 1>
  </Colon A>

  <Colon B>
    <Element 0>
      <Caption>Answer 1</Caption>
    </Element 0>
    <Element 1>
      <Caption>Answer 2</Caption>
    </Element 1>
  </Colon B>

</Test 1>
</Test with colons 1 - Example>
```

Face recognition

The software allows you to log into admin after there is a facial recognition + facial classification. Both the recognition feature and the classification, face is perfectly functional, but has real-time recognition problem(it have a 1 frame/second). However the desired goal was achieved, implementation of these concepts in educational software. Mentioned that the facial recognition, classification and the recognition, is completely written by me and presented in other competitions. I have not used any SDK.



Likeness between Logicus and a Compiler

Logicus uses many concepts of a compiler and compilation theory:

- Analyze syntactic search words how do not belong to vocabulary
- Create tables of variables
- Have a lexical analysis - undeclared variables, verify the correctness of an expression.
- Use lexems, tokens.
- Like compiler, Logicus creates the source code translation scheme. Compilers translate source code to assembly. Codes are equivalent. The both are translators.
- Create a associated graph how is equivalent with logical scheme. Compiler creates an associated graph with the source code.

Because the dream of any developer, is: an application to keep in touch with the all users, i introduced a small update, not only for executables, and for examples. So that, I can publish to user a a wide range of examples to use software .

Update

Host: <http://www.neuroslab.com/Downloads/Update/Logicus 2.0/>

Check for updates

List news | Information - report

	read and write array.xml	Size: 11695 bytes Date: 19.06.2010
	hello world.xml	Size: 1493 bytes Date: 18.06.2010
	<input checked="" type="checkbox"/> Download 10.xml	Size: 121 bytes Date: 27.06.2010
	<input checked="" type="checkbox"/> Download referat1.doc	Size: 140288 bytes Date: 27.06.2010

Generals

Number of files: 8
Not updated files: 3
Updated files: 5
Total files size: 3556321bytes

Download and install

Update

Host: <http://www.neuroslab.com/Downloads/Update/Logicus 2.0/>

Check for updates

List news | Information - report

Updating status

The progress of updating

Downloaded: 2/2

Downloaded and installed files			
Name	New Size	Path	Type
10.xml	121	C:\Documents and Settings\nodsoftware\Desktop\Logicus2 AI2\Edu\Lesson...	
referat1.doc	140288	C:\Documents and Settings\nodsoftware\Desktop\Logicus2 AI2\Edu\Lesson...	

Information

Finish updated! You Must restart application

OK

As software programming means a very efficient management of errors I enter, to program the concept of Debug Mode which displays a console evolution about code execution. It can be very useful in debugging the program, because, classical methods like Watch is no longer efficient

Colored with yellow lines are good features of software written with red lines are malfunctioning.

```

Log Form
Open Logicus - 01:22:35...ok
Open Into Form...ok
Close into form...ok
Tray to create myDraw object...ok
Init DrawClass instance DrawBox...ok
Try to show agent...ok
Succesfull show Agent...ok
Tray to load scheme from: C:\Documents and Settings\hez\Desktop\Logicus2\Save
\for_2.lgs...ok
New scheme...ok
Object added to scheme container Start11 type Start...ok
Object added to scheme container Input11 type Read...ok
Object added to scheme container Atribuire11 type Atribuire...ok
Start to download updates. There are 1 files to download && install...ok
Tray to download: expresie_complicata and replace after: C:\Documents and
Settings\hez\Desktop\Logicus2\Examples\expresie_complicata.lgs...ok
Start to download updates. There are 1 files to download && install...ok
Tray to download: nopuri_multe and replace after: C:\Documents and Settings
\hez\Desktop\Logicus2\Examples\nopuri_multe.lgs...ok
  
```

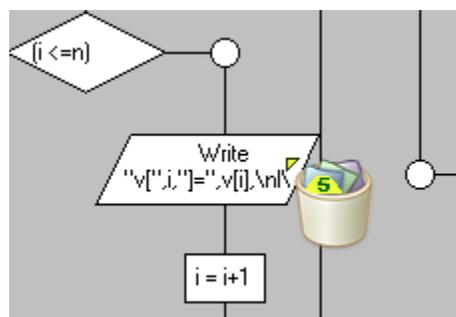


Magnify - button, with "magnifier" icon can display, some basic information on selected objects.

Informatii despre obiect	
Name	If4
Type	If
X	251
Y	316
Width	80
Height	40
Brush Color	c\White
Pen Color	c\Black
Text Color	c\Black
Yes Next	If_yes17
No Next	If_no16
Instruction	(i < n)



Recycle bin – button with "trash" icon can delete objects from the current scheme created by the user.



Algorithms

1. Graph theory - Schemes were represented as graphs when were executed
2. Proorder search in binary tree.
3. Recursion
4. Divide et impera

Used technique:

1. Canvas
2. Double buffered
3. Threads(Simulating, update, and in about)
4. Thread was synchronized
5. All of the objects and variables was dynamicall allocated.
6. Update
7. Change language I had use recursion for algorithm switching languages.

Used tehnologies

1. Borland Delphi 2009
2. Borland Delphi 6
3. BusinessSkin
4. NeurosLab(software how was writed by me and had been presented to competitions)

Bibliography

For documentation, I had studied the next authors.

- Retele neuronale –arhitecturi si algoritmi Virgil Tiponut, Catalin Caneanu Universitatea de Vest
- Retele neuronale Aplicatii -Virgil Tiponut Catalin Daniel Caneanu. Universitatea. de Vest
- Retele neuronale - Nicolae Tandareanu, Universitatea din Craiova.
- Inteligenta artificiala- Razvan Andonie,Universitatea Bucuresti

- **ACM Digital Library**

© CopyRight 2009 by Giga-Software (www.giga-software.com)

Budisteanu Ionut Alexandru (aka. h33z0r)

Student 10 class grade at High School „Colegiul National Mircea cel Batran”

About me and my awards <http://neurosab.com/Downloads/Doc/CV/>

(open in a new tab page)



For further queries related to this project contact us through the Giga-Software website at www.giga-software.com or you can e-mail us at: ibudisteanu@acm.org , ibudisteanu@giga-software.com

2010-2011 **Ionut Alexandru Budisteanu**
Grup Scolar Olthim

www.giga-software.com
www.seewithtongue.info
ibudisteanu@acm.org

Ferdinand Street no 22, Ramnicu Valcea, 240156, Valcea, Romania