GryphonSharp Proposal

a.sukachev@lancaster.ac.uk

Coordinator: Abe Karnik (a.karnik@lancaster.ac.uk)
Research group: MSc410 (Human Computer Interaction)

0.1   introduction

Recently, world has been heavily digitized - practically, every aspect of everyday life is in digital form ranging from digital watches to smart houses. With the increasing demand for programming in nearly all science disciplines, need for programming education has risen drastically [4, 2, 6]. Since introduction of Visual Programming environments such as Scratch into education, average performance of students has increased [5] while barriers of getting intuition in programming fell down [3]. There are a lot of languages out there, which provide constructs and other tools to achieve a given task.

0.2   Background

0.3   Scope

Project will cover basic programs without extensive programming features, such as attributes, abstraction or even basic objects. The aim here is to fuse already existing studies of visual programming languages into a working prototype that is capable of producing simple programs, such as guess a number or console calculator.

0.4   Application

Since the invention of C language, computer languages have been tremendous help in many applications. General purpose languages emerged to put extra layer of abstraction on top of "one-at-the-time instruction" assembly language.[1] As mentioned above by [3, 5], the final prototype of the language can be used in educational purposes

0.5   Methodology

Overview

Implementing fully functioning programming language that compiles into assembler is a genuinely time-consuming task, which also requires in-depth knowledge of different processor architectures such as ARM, AMD64 and Intel64. For this reason, the project's visual language instead will transpile into C# Intermediate Language (henceforth IL) using CodeDOM feature of the language.

Visual Editor

The visual editor, or in this case node editor, will be implemented as a webpage canvas using Konvajs framework. Konvajs lets you easily serialize and deserialize data and populate the 'code map'. Refer to figure 1.
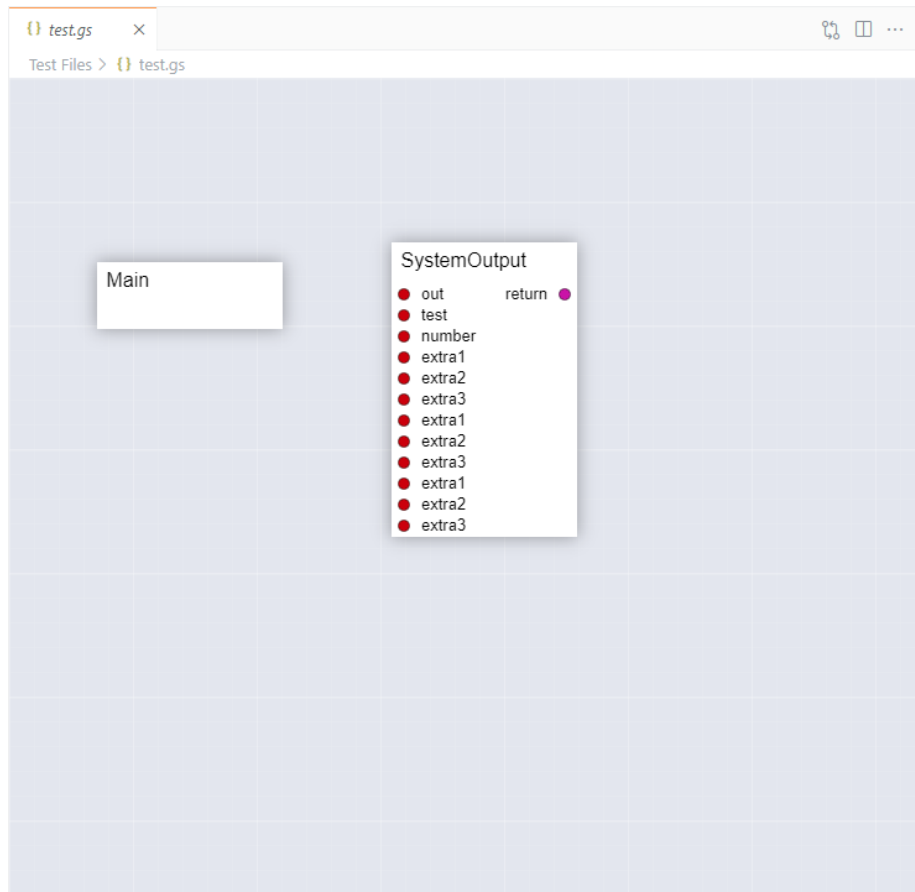
Figure 1: Code Map

The webpage will be able to serialize all node structure into a JSON file, which will be a source code of a file. This JSON can then be parsed by Transpiler and convert its contents into C# IL. The Konvajs framework will be closely tied with Electron app called 'Visual Studio Code' (henceforth VSCode). VSCode

Transpiler

As previously stated, the primary purpose of a transpiler is converting GryphonSharp-generated JSON files into C# IL. Transpiler is named

Language Server

0.6 Time Plan

Project's time plan will span over 3 months and is represented by Gantt Chart below.

0.7 Summary

Bibliography

[1] Jan V. Garwick. Programming languages: Gpl, a truly general purpose language. *Commun. ACM*, 11(9):634–638, September 1968.

[2] Milica Jovanović, Jasmina Dlačić, and Milan Okanović. Digitalization and society's sustainable development -measures and implications.

[3] Caitlin Kelleher and Randy Pausch. Lowering the barriers to programming. *ACM Computing Surveys*, 37:83–137, 06 2005.

[4] Jane Margolis, Joanna Goode, and David Bernier. The need for computer science. *Educational Leadership*, 68(5):68–72, February 2011.

[5] Chun-Yen Tsai. Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy. *Computers in Human Behavior*, 95:224–232, 06 2019.

[6] L. A. Vaca-Cárdenas, F. Bertacchini, A. Tavernise, L. Gabriele, A. Valenti, D. E. Olmedo, P. Pantano, and E. Bilotta. Coding with scratch: The design of an educational setting for elementary pre-service teachers. In *2015 International Conference on Interactive Collaborative Learning (ICL)*, pages 1171–1177, 2015.