|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SEMESTER** | | **Fall-2022** | **CLASS AND SECTION** | | **BSCS-5** |
| **TITLE OF PROJECT** | | | | | |
| **Student compiler** | | | | | |
| **Group Members** | | | | | |
| **email:** | **aligauharrrrag@gmail.com** | | | **Ali Gauhar** | |
| **EXECUTIVE SUMMARY OF PROJECT PROPOSAL** | | | | | |
| The idea of the project is to construct a compiler which benefit the primary student who cannot learn the difficult concept of the languages like C++ and java, like in our time when we were kids, we use to work on GWBASIC but since its concepts are absolute now so we thought of making a new language that can help the beginners learn and grab the concepts easily. The compiler has been programmed in Python. It is designed for newbies or the beginners. Data types and syntax is unique and totally self-constructed or developed. The student can easily get the output without involving in the detailed logic and syntax. This project facilitates the student to make interest in the field of programming. The student will be able to understand the basic of the programming easily and there is a lot of steps which will not be necessary to perform like saving file, declaration of variable or store the output in another variable. Like I knew when we were in first semester, we me and my friends faced an error and we googled it every where every time it gave a different solution and then at last, we got to knew that we were not saving the file before compiling therefore the previous code was not being aligned with the new one. | | | | | |
| **PROJECT PURPOSE, SCOPE AND OBJECTIVES** | | | | | |
| **PURPOSE:**  The purpose of this compiler is to help the primary student of school or the student would be able to play with coding without being lost into logic. The student will be able to learn languages more easily. It will help them get better with the basics of program and they can play around with their output by just writing similar English peace of code which is near to human language mostly and has a simple structure of loop. Without taking much care about the structure, they can learn to code.  we faced a lot of problem in school time in programming because we were not familiar with keywords and structure of program, so to solve this kind of problem. we designed a compiler to reduce the complexity of program in order of keeping simple, and easy words and make the structure of program easy.it also give relevant errors when the coder write any irrelevant thing or make any mistake which is very easy to understand and not like other languages which shows error that are unable to understand for beginners.  **SCOPE:**  Compilation techniques is essential for understanding how programming languages and computers hang together. Technology is growing exponentially. Computer programming is vital today because so much of our world is automated. People need to be able to control the interaction between humans and machines. Since computers and devices can do things so efficiently and accurately, we use computer programming to harness that computing power. People know what they want but they have no clue how it would be done. As the main idea is to develop the coding skills instead of focusing much on the language specifications and syntax. This project will be guide the aspirant to achieve his goal easily and help the user to develop their skill in the easy way.  **OBJECTIVES:**   * To build interest in school children * To get better understanding of programming easily * Translation of a higher-level language into the fastest possible machine language that accurately represents the high-level language source. It is an approach to increase the coding skills instead of focusing a lot on the syntax and facing difficult keyword to remember. A user-friendly compiler, that is easy to use and making updates and adding up new things to it with time according to the requirements would be easier. To facilitate the students in better way to understand the programming. It will attract more users because the complexity will be reduced and when things become easy then everyone wants to learn as this is the psychology of human being. | | | | | |
| **PROJECT DESCRIPTION (most important part)** | | | | | |
| This project will solve the problem of primary student of school and give easy method to get the output and clear their concept related to programming failed. Now a days everyone wants to make their concept strong in programming and they face a lot of difficulties due to not have strong basic fundamental concept related to programming. This project aims to facilitate the beginner. The student can develop their programming skills using this compiler  This compiler hides the complexity of program by interpreting the code line by line and handle many exceptions. Our compiler will act as an interpreter simple means interpreter over interpreter which is amazing because it facilitated the user to made error and the compiler will handle it and tell them the relevant solution.   1. Lexical Analyzer 2. Parser or Syntax Analyzer 3. Intermediate Code generator 4. Code Optimization 5. Code Generation   **1.Lexical Analyzer:** It will take each line of code and spit it into chunks or basic units of data. All the chunks or piece of information has some meaning or definition to it. It is labeled with the identifier or definition, creating tokens. Each token is stored in a table called ‘Symbol Table’. Then all the tokens are passed on to the parser to check the syntax. It uses Regular expression to identify the basic units and can be token sized.  **2.Parser:** The parser will now take the tokens, and check the CFG (Context Free Grammar) to check if the syntax is correct. It creates a parse tree. It basically checks if the pieces of the puzzle are put in right place. Once the syntax is checked, the code is handed over to the scanner.  **3.Intermediate Code generator:** Intermediate code can translate the source program into the machine program. Intermediate code is generated because the compiler can't generate machine code directly in one pass. Therefore, first, it converts the source program into intermediate code, which performs efficient generation of machine code further.  **4.Code Optimization:** The code optimization in the synthesis phase is a program transformation technique, which tries to improve the intermediate code by making it consume fewer resources (i.e. CPU, Memory) so that faster-running machine code will result.  **5.Code Generation**: In computing, code generation is part of the process chain of a compiler and converts intermediate representation of source code into a form (e.g., machine code) that can be readily executed by the target system. Sophisticated compilers typically perform multiple passes over various intermediate forms. Tasks of Compiler The main tasks performed by the Compiler are:   * Breaks up the up the source program into pieces and impose grammatical structure on them * Allows you to construct the desired target program from the intermediate representation and also create the symbol table * Compiles source code and detects errors in it. * The main task of a compiler is to map programs written in a given source language into a target language * Often, the source language is a programming language and the target language is a machine language * Some exceptions: Source-to-source translators, machine-code translation, data manipulation in XML * Part of the task of a compiler is also to detect, whether a given program conforms to the rules of the source language. * A specification of a compiler consists of   + A specification of its source and target languages   + A specification of a mapping between them of machine | | | | | |
| **TEAM PROFILE** | | | | | |
| Ali Gauhar: Working on the grammar and coding of laxer, interpreter, etc. | | | | | |

|  |
| --- |
| **ASSUMPTIONS AND CONSTRAINTS** |
|  |
| **Assumptions:**  we will try to implement as many functions as we can and make this project more intractable.  **Constraints:**  In this project we have lot of coding to do, so we will try to do best and add new functions and make this project more intractable and good.  **SOFTWARE USED:**  Vs code (language python)  **HARDWARE RECOMMENDATION**:  10 MB Free Space  4 GB RAM  **Language specification**:  Python used for programming |
| **PROJECT DELIVERABLES NOT CHANGEABLE** |
| **Deliverables include**   * Software Project Proposal. * Project progress * Project report * Team member’s work, as per their contribution, you should have to be honest with your future. |
| Time Line |
| |  |  |  | | --- | --- | --- | | **ITEM** | **DATE** | **SIGNATURE** | | Project Proposal | 06-05-2022 |  | | Project Submission |  |  | |

|  |
| --- |
| **For Teacher Use Only** |

|  |
| --- |
| **REMARKS** |
|  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course Teacher:** |  | **Signature** |  | **Date:** |  |
| **Lab Teacher:** |  | **Signature:** |  | **Date:** |  |