|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SEMESTER** | | | **Fall-2022** | **CLASS AND SECTION** | | **BSCS-5** |
| **TITLE OF PROJECT** | | | | | | |
| **Basic C++ Compiler** | | | | | | |
| **Group Members** | | | | | | |
| **S#** | **Student(s) Name** | | | | **Enrollment Number** | |
|  | **Muhammad Ahsan Arshad Manzoor** | | | | **02-134202-087** | |
| **email:** | | **mahammadahsan@gmail.com** | | |
| **S#** | **Student(s) Name** | | | | **Enrollment Number** | |
|  | **Aliyan Rehman** | | | | **02-134202-052** | |
| **email:** | | **Aliyanrehman33@gmail.com** | | |
| **S#** | **Student(s) Name** | | | | **Enrollment Number** | |
|  | **Usama Ashraf** | | | | **02-134202-044** | |
| **email:** | | **Usamaalee33@gmail.com** | | |
| **EXECUTIVE SUMMARY OF PROJECT PROPOSAL** | | | | | | |
| The idea of the project is construct a compiler which can compile the new language called basic C++ which is based on C++. The compiler is programmed in Python. It is designed for the beginners. The code to be generated will be in form of Intermediate code and compiler will have lexical, syntax and semantic analyzer. Since, Python provide many built-in features and functions, so it would be easy to implement. The code can be stored in a text file or the user can directly enter the code and its intermediate code will be generated in file. | | | | | | |
| **PROJECT PURPOSE, SCOPE AND OBJECTIVES** | | | | | | |
| **PURPOSE:**  The purpose of this compiler is to help the beginners to learn C++ more easily. It will help them get better with the basics of C++ by understanding basic structures of C++ Without taking much care about the Difficult structures and complex library inclusions, they can learn to code.  I got this is idea when some of the students not coming from computer background approached me, they wanted to learn C++ basics but were confused by library and function callings. So, it is an easy and reliable solution for them.  **SCOPE:**  Since, for non-programmers programming is a big deal. But with the development and progress of time technology and computer is becoming a basic need for everyone. 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, it will help a large range of users to develop their coding skills and learn C++. It is easy to modify and with time by making updates according to the user use, it will attract more users.  **OBJECTIVES:**  The objective of the compiler is very clear and as discussed earlier, it is an approach to increase the coding skills instead of focusing a lot on the syntax. A user-friendly software, that is easy to use and making updates and adding up new things to it with time according to the requirements would be easier. The main idea is to develop the coding skills instead of focusing much on the language specifications and syntax, it will help a large range of users to develop their coding skills and learn C++. It is easy to modify and with time by making updates according to the user use, it will attract more users. | | | | | | |
| **PROJECT DESCRIPTION (most important part)** | | | | | | |
| An Basic C++ Compiler, as the name suggests, is a program developed to compile the code written in Basic C++. A Basic C++ Compiler is lightweight software developed with the sole purpose of compiling code written in Basic C++. Basic C++ will significantly help amateurs and non-professionals to accomplish Basics of C++ quickly and successfully.  It has multiple layers or we say steps to convert a code file completely convert into Intermediate code. The error is checked on each part or layer. The sections include:   1. Lexical Analyzer 2. Syntax Analyzer 3. Semantic Analyzer   **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 tokenized.  **Syntax Analyzer:** The Syntax Analyzer 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 Semantic Analyzer.  **Semantic Analyzer:** Now the parse tree is checked for the semantics. The scanner is a subroutine which is frequently called by an application program like a compiler. The primary function of a semantic analyzer is to check semantics errors in code  There is a symbol table which stores all the relevant information to get the desired output. Each section or part has its own errors. If the errors are not detected properly in the beginning stage, it may affect the output. So, all the possible errors and constraints are dealt at that stage. 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 * Manage storage of all variables and codes. * Support for separate compilation * Read, analyze the entire program, and translate to semantically equivalent * Translating the source code into object code depending upon the type of machine | | | | | | |
| **TEAM PROFILE** | | | | | | |
| Muhammad Ahsan, I am doing the coding and planning.  Aliyan Rehman, I’m doing documentation and coding.  Usama Ashraf, I’m doing planning and coding. | | | | | | |

|  |
| --- |
| **ASSUMPTIONS AND CONSTRAINTS** |
|  |
| **Assumptions:**  We will try to implement as many functions as we can and make this project more interactable.  **Constraints:**  In this project we have lot of coding to do, and less time due to exams assignments and quizzes etc.  So we will try to do best and add new functions and make this project more interactable and good.  **SOFTWARE USED:**  Visual Studio 2022  **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 | 04-12-2022 |  | | Project Submission |  |  | |

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

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

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