**INHA UNIVERSITY TASHKENT**

**DEPARTMENT OF CSE & ICE**

**FALL SEMESTER 2019**

**SOC4140 - THEORY OF PROGRAMMING LANGUAGES**

**TERM PROJECT PROPOSAL**

PROJECT TITLE[ENTER HERE]

**Submitted by**

**Student Names Student ID**

**Group : Junior**



|  |
| --- |
| **Term Project Guidelines**  **TERM PROJECT INVOLVES THE FOLLOWING MAIN TASKS :**   * **DESIGN OF YOUR OWN PROGRAMMING LANGUAGE**   **You are required to propose a new programming language considering the following factors into account :**   * **Language Evaluation Criteria – Readability, Writability, Simplicity, Orthogonality, Reliability, Cost, Portability leading to ease of coding and Program maintenance** * **Different Applications Domains – Scientific, Business, Artificial Intelligence, Systems Programming, Web, Graphics and other application areas** * **Different Programming Language Paradigms – Imperative, Object Oriented, Declarative, Functional, Logical, Markup and taking the best constructs and features from these language paradigms** * **Above all, considering your experience of developing applications for different domains using different languages during the last three years and feeling the need to have a better programming language with easier constructs and best features which make the user to write programs for different applications (or application of his/her choice) with ease of learning and coding with minimum time and effort** * **Prepare complete specification for all the constructs/features which are going to be incorporated in your proposed Language.** * **DESIGN AND IMPLEMENTATION OF LEXICAL ANALYSER USING FLEX** * **Prepare proper Grammar specification for all the constructs/features of your proposed programming language using BNF/EBNF** * **Implement Lexical Analyser using FLEX** * **DESIGN AND IMPLEMENTATION OF SYNTAX ANALYSER/PARSER USING BISON** * **Implement Parser for your grammar using BISON** * **DESIGN AND IMPLEMENTATION SEMANTIC ANALYSER & INTERMEDIATE CODE GENERATOR WITH MACHINE INDEPENDENT OPTIMIZATION**    + **Design & Implement a Semantic Analyser for the AST output provided by the Parser**   + **Design & Implement an Intermediate Code Generator considering a hypothetical machine**   + **You can write your own Intermediate Code Generator or use any of the tools available to generate the Intermediate Code.**   + **After generating the Intermediate Code, perform machine Independent Optimization on the code generated** * **DESIGN & IMPLEMENTATION OF CODE GENERATOR WITH MACHINE DEPENDENT OPTIMIZATION (IN CASE OF COMPILER IMPLEMENTATION) OR**   **DESIGN & IMPLEMENTATION OF VIRTUAL MACHINE FOR INTERMEDIATE CODE INTERPRETATION (IN CASE OF INTERPRETER IMPLEMENTATION)**   * **Here you have two choices to make – either designing a compiler or a Hybrid Interpreter** * **In the case of Compiler, you need to design and Implement a code generator converting Intermediate code to Intel x86-64 machine Instructions** * **In the case of a Hybrid Interpreter, you need to design and Implement a Virtual Machine to Interpret the Intermediate code and produce the results** * **PROJECT TESTING PHASE – WRITING DIFFERENT PROGRAMS USING YOUR PROPOSED PROGRAMMING LANGUAGE AND EXECUTING THEM USING YOUR OWN COMPILER OR INTERPRETER**   + **This is the Compiler/Interpreter Testing phase where you will write different programs for various applications using your own proposed language and Compile or interpret using your own Compiler/Interpreter**   + **To enter your program, you can use the existing editors or you can design an Integrated Design Environment(IDE) very similar to Python IDLE or Microsoft/Borland IDE or any other IDE of your choice with the following features(YOU WILL GET ADDITIONAL POINTS FOR THIS PART):**     - **Create a new file, open an existing file, file editing features, File save & Exit**     - **In the case of Interpreter, provide an Interpreter shell environment where user can enter the program statement on a line and interpret it to get the results (line by line Interpretation)**     - **In the case of Compiler, provide a menu option Compile & Run so that you can pop up a new window showing compilation your program and the Results**     - **You can also add several other features available on most of the IDEs** * **EACH PROJECT TEAM CAN CONSIST OF 4 TO 8 MEMBERS) WITH A MAXIMUM OF 15 PROJECT TEAMS IN A SECTION.** * **THE PROPOSED TITLE OF THE TERM PROJECT (i.e., NAME OF THE YOUR PROPOSED PROGRAMMING LANGUAGE) ALONG WITH ABSTRACT, BRIEF PROJECT DESCRIPTION, REQUIREMENTS DEFINITION DOCUMENT IN THE FORM OF A PROJECT PROPOSAL (PRELIMINARY DOCUMENT) MUST BE SUBMITTED BY MONDAY 11TH NOVEMBER 2019.** * **PREPARE THE PROJECT PROPOSAL USING THE TEMPLATE PROVIDED AT THE ARCHIVE SECTION OF THE ECLASS PORTAL.** * **ONE HARD COPY OF THE PROJECT PROPOSAL OF EACH TEAM SHOULD BE HANDED IN AT THE OFFICE BY THE TEAM LEADER.** * **EVERY MEMBER OF THE TEAM MUST UPLOAD THE SOFTCOPY OF THE PROJECT PROPOSAL AT THE E-CLASS PORTAL** * **LAST DATE FOR SUBMISSION OF THE PROJECT PROPOSAL IS 11TH NOVEMBER 2019** * **PROJECT PROPOSAL PREPARED USING THE TEMPLATE WILL ONLY BE ACCEPTED** * **LATE SUBMISSIONS ARE NOT ENTERTAINED, ADHERE TO THE DEADLINE STRICTLY** |

**TITLE OF THE PROJECT:**

[Provide the Title here]

**PROBLEM STATEMENT :**

[State clearly the statement of the problem which you are going to tackle]

**ABSTRACT :**

[Provide a gist or executive summary of the project you are going to carry out providing all the details in brief]

**PROJECT OVERVIEW :**

* [Briefly Describe how the following tasks are going to be performed : design of your own programming language, design and implementation of lexical analyser using flex, design and implementation of syntax analyser/parser using bison, design and implementation semantic analyser & intermediate code generator with machine independent optimization, design & implementation of code generator with machine dependent optimization (in case of compiler implementation) or design & implementation of virtual machine for intermediate code interpretation (in case of interpreter implementation), project testing phase – writing different programs using your proposed programming language and executing them using your own compiler or interpreter. ]

**PROJECT ACTION PLAN :**

**Term Project Tasks Breakup Weekwise**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No** | **Duration** | **Specify Tasks to be carried out** | **Team Members assigned to do various components of this task** |
| 1. | 10th Week  11th Nov. to 17th Nov. 2019 | **Design of our own programming language** |  |
| 2. | 11th  Week  18th Nov. to 24th Nov. 2019 | **Design and implementation of lexical analyzer using flex**  **Design and implementation of syntax analyzer/parser using bison** |  |
| 3. | 12th Week  25th Nov. to 1st Dec. 2019 | **Design and implementation semantic analyzer & intermediate code generator with machine independent optimization** |  |
| 4. | 13th Week  2nd Dec.to 8th Dec. 2019 | **Design & implementation of code generator with machine dependent optimization (in case of compiler implementation) or**  **Design & implementation of virtual machine for intermediate code interpretation (in case of interpreter implementation)** |  |
| 5. | 14th Week  8th Dec. to 11th Dec. 2019 | **Project testing phase – writing different programs using your proposed programming language and executing them using your own compiler or interpreter**  **Term project final report preparation** |  |
| 6. | 14th Week  12th Dec. to 14th Dec. 2019 | * **Term Project Final Report Submission** * **Term Project Presentation** | **ALL** |

**PROJECT TEAM :**

**Team Leader :**

**Name Student ID Signature with date**

**1.**

**Team Members :**

**Name Student ID Signature with date**

**2.**

**3.**

**4.**

**5.**

**6.**

**7.**

**8.**

**REFERENCES :**

[Provide the list of the books, library resources, websites referenced for understanding various topics required for the design and implementation of your term Project]