|  |  |  |
| --- | --- | --- |
| Course: | **Compiler Construction** | Date: 5th January 2024 |
| Course Code: | **CSC 323** | Session: I |
| Faculty’s Name: | Qazi Haseeb Yousaf | Max Marks: 30 |
| Time Allowed: | 60 Minutes | Total Pages: 1 |

Student’s Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Enroll No: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(USE CAPITAL LETTERS)

**Question # 1 (10 Marks) (CLO3)**

Consider the following CFG:

**S 🡪 SA | aB | b**

**A 🡪 Ab | ε**

**B 🡪 Sd | c**

Given the above CFG**, Generate** a CFG that does not have Left Recursion [10 marks]

**Question # 2 (10 Marks) (CLO3)**

#include <iostream>

using namespace std;

int main() {

int a = 3, b = 6;

int **c** = a+b;

int **d**

string **e** = a+b;

d = (a-b)\*c+(a+c)/b

return 0;

}

1. **Generate** Intermediate code for the above code.
2. **Generate** the parse tree for the above code

**Question # 3 (10 Marks) (CLO2)**

1. **Write** a small lexical analyzer code which recognizes the identifiers, numbers, brackets, semicolons, special characters and returns appropriate tokens

**Best of Luck**