|  |  |
| --- | --- |
| **Department of Computer Science and Engineering**  Quiz Exam Questions  CSE 420 Compiler Design  Time 30 minutes CSE411 Section 05 Total marks 20 | |
| Q1. Why do study compiler design? Write down two compiler and interpreters name. Do you think compiler is one kind of interpreter, justify your answer? | Marks 1+2+1=4 |
| Q2. Write four phases of compiler and write short note about each of them. | Marks 4 |
| Q3. a) Write regular expressions for the following program in ??? position.  **public** **class** **ValidateDemo** {  **public** **static** **void** main(**String**[] args) {  **List**<**String**> input = **new** **ArrayList**<**String**>();  input.add("123-45-6789");  input.add("9876-5-4321");  input.add("987-65-4321 (abcd)");  input.add("987-65-4321 ");  input.add("192-83-7465");  **for** (**String** ssn : input) {  **if** (ssn.matches("???")) {  **System**.**out**.println("Found good SSN: " + ssn);  }  }  }  }  b) Write output for the following program  import java.util.regex.\*;  class RegexExample{  public static void main(String args[]){  System.out.println(  Pattern.matches("[Tt]om", "Tom")); \\True/False  System.out.println(  Pattern.matches("[tT]im|[jJ]in", "Tim")); \\True/False  System.out.println(  Pattern.matches(".\*abc.\*", "deabcpq"));  System.out.println(  Pattern.matches("^[^\\d].\*", "123abc"));  **Department of Computer Science and Engineering**  Quiz Exam Questions  CSE 420 Compiler Design  Time 30 minutes CSE411 Section 05 Total marks 20  System.out.println(  Pattern.matches("^[^\\d].\*", "abc123"));  System.out.println(  Pattern.matches("[a-zA-Z][a-zA-Z][a-zA-Z]", "apZx"));  System.out.println(  Pattern.matches("\\D\*", "abcde123"));  System.out.println(  Pattern.matches("^This$", "This is Chaitanya"));  String content = "This is Chaitanya " +  "from Beginnersbook.com.";  String pattern = "B.\*book.\*";  boolean isMatch = Pattern.matches(pattern, content);  System.out.println("The text contains 'book'? " + isMatch);  }  } | Marks 3  Mark 9 |

**Solution**

import java.util.regex.\*;

class RegexExample{

public static void main(String args[]){

// It would return true if string matches exactly "tom"

System.out.println(

Pattern.matches("tom", "Tom")); //False

/\* returns true if the string matches exactly

\* "tom" or "Tom"

\*/

System.out.println(

Pattern.matches("[Tt]om", "Tom")); //True

System.out.println(

Pattern.matches("[Tt]om", "Tom")); //True

/\* Returns true if the string matches exactly "tim"

\* or "Tim" or "jin" or "Jin"

\*/

System.out.println(

Pattern.matches("[tT]im|[jJ]in", "Tim"));//True

System.out.println(

Pattern.matches("[tT]im|[jJ]in", "jin"));//True

/\* returns true if the string contains "abc" at

\* any place

\*/

System.out.println(

Pattern.matches(".\*abc.\*", "deabcpq"));//True

/\* returns true if the string does not have a

\* number at the beginning

\*/

System.out.println(

Pattern.matches("^[^\\d].\*", "123abc")); //False

System.out.println(

Pattern.matches("^[^\\d].\*", "abc123")); //True

// returns true if the string contains of three letters

System.out.println(

Pattern.matches("[a-zA-Z][a-zA-Z][a-zA-Z]", "aPz"));//True

System.out.println(

Pattern.matches("[a-zA-Z][a-zA-Z][a-zA-Z]", "aAA"));//True

System.out.println(

Pattern.matches("[a-zA-Z][a-zA-Z][a-zA-Z]", "apZx"));//False

// returns true if the string contains 0 or more non-digits

System.out.println(

Pattern.matches("\\D\*", "abcde")); //True

System.out.println(

Pattern.matches("\\D\*", "abcde123")); //False

/\* Boundary Matchers example

\* ^ denotes start of the line

\* $ denotes end of the line

\*/

System.out.println(

Pattern.matches("^This$", "This is Chaitanya")); //False

System.out.println(

Pattern.matches("^This$", "This")); //True

System.out.println(

Pattern.matches("^This$", "Is This Chaitanya")); //False

}

}

**public** **class** **ValidateDemo** {

**public** **static** **void** main(**String**[] args) {

**List**<**String**> input = **new** **ArrayList**<**String**>();

input.add("123-45-6789");

input.add("9876-5-4321");

input.add("987-65-4321 (attack)");

input.add("987-65-4321 ");

input.add("192-83-7465");

**for** (**String** ssn : input) {

**if** (ssn.matches("^(\\d{3}-?\\d{2}-?\\d{4})$")) {

**System**.**out**.println("Found good SSN: " + ssn);

}

}

}

}