DHARMSINH DESAI UNIVERSITY, NADIAD
FACULTY OF TECHNOLOGY
B.TECH. SEMESTER VI [IT]
SUBJECT: (IT608) LANGAUGE TRANSLATOR

Examination	**:First Sessional**	**Seat No.**	**: _____**
Date	**: 07/01/2013**	**Day**	**: Monday**
Time	**: 12.00 to 1.15**	**Max. Marks**	**: 36**

INSTRUCTIONS:
1. Figures to the right indicate maximum marks for that question.
2. The symbols used carry their usual meanings.
3. Assume suitable data, if required & mention them clearly.
4. Draw neat sketches wherever necessary.
5. ^ indicates null symbol.

Q.1 Do as directed.

(a) Translator for low level programming language were termed as **[1]**
(A) Assembler (B) Compiler (C) Linker (D) Loader

(b) A linker program **[1]**
(A) Places the program in the memory for the purpose of execution.
(B) Relocates the program to execute from the specific memory area allocated to it.
(C) Links the program with other programs needed for its execution.
(D) Interfaces the program with the entities generating its input data.

(c) The number of tokens for punctuation in the following C statement **[1]**
printf("i=%d, &i= %x", i, &i);
is (A) 4 (B) 8 (C) 10 (D) 2

(d) Match the following appropriately. **[2]**

A. Brute force	1) Multiple derivations.
B. Recursive descendent parser	2) Preferred when language does not permit Recursion.
C. LL parser	3) No backup needed.
D. Ambiguity	4) Parses string with all possible combinations.

(e) FLEX is a _____generator tool.(Parser, Linker, Scanner, Loader) **[1]**

(f) Analysis phase is _____ specific. (Platform, Language, Tool) **[1]**

(g) Parse the string "abdbdcde" using brute force method. G1 : S→aABe A→bdA |c| de B→d **[3]**

(h) What is Back patching? Explain with example. **[2]**

Q.2 Attempt ***Any Two*** from the following questions. **[12]**

(a) For the following 'C' fragment, identify and list the lexemes that make up tokens. **[6]**

```
#define<stdio.h>
main () { int m, x; /*a simple program for finding*\max of two*/numbers*/
        printf ("enter two numbers"); m=23; x=4.54; fi(x>m);
        printf ("max is y"); else printf ("max is z"); }
```

(b) Write RDP for the following grammar G2 : S→Xb | aM |cS X→d|e M→y|z **[6]**

(c) Draw transition diagram for the following rules. **[6]**
Whitespaces- blanks, tabs, newlines. **Keywords**- if, else, then, end, start, while, for.
User identifiers- start with Roman letters, followed by one or more alphabets.
Character constants- any thing between ` `. Apostrophe can be included in the character constant, but it should occur in pair. Ex- `it``s valid statement`.

Q.3 (a) Determine following grammar is LL (1) or not. **[8]**
Grammar G3: E→T+E|T T→int|int*T|(E)
If it is LL (1), draw parse table and parse the string "int*int+int".
Else explain the reason why it is not LL (1).

(b) Is the following grammar is ambiguous? Justify with the input string "00111". **[4]**
Grammar G: S→AS|^ A→A1|0A1|01.

OR

Q.3 (a) Determine following grammar is LL (1) or not. **[6]**
S→AaAb | BbBa A→^ B→^

(b) Write short note on Linker and Loader. **[6]**