**❶** MIDTERM TEST

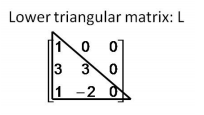
**COMPILER CONSTRUCTION**

(70 minutes- Open book- No Laptop)

(70 phút – được sử dụng tài liệu giấy, không được dùng các loại máy tính)

**Note:** The functions you are required to build are parts of project Parser used in practice exercise No 2. You must use the data structures declared in project Parser.

1/ Matrix in A lower triangular matrix is a square matrix which all entries above the main diagonal are zero. For example,

  
Write a KPL program to read elements in a matrix and check whether the matrix is lower triangular matrix or not. The program prints 1 if the matrix is lower triangular, otherwise it prints 0.

2/ Given the set of syntax rules:

**22) FunDecl ::= KW\_FUNCTION TK\_IDENT Params SB\_COLON**

**BasicType SB\_SEMICOLON Block SB\_SEMICOLON**

**23) ProcDecl ::= KW\_PROCEDURE TK\_IDENT Params**

**SB\_SEMICOLON Block SB\_SEMICOLON**

**24) Params ::= SB\_LPAR Param Params2 SB\_RPAR**

**25) Params ::= ε**

a. Prove that productions from 24 to 25 satisfy LL(1) condition.

b. Write function void compileParams(void) satisfying the above syntax rules. Assuming functions compileParams2 and compileParam have been built.

3/ Write function **void readIdentKeyword (void)** satisfying the following rules:

* Identifiers are made up of letters, digit and underscore (ASCII code 95); the first character must be a letter.
* [Uppercase and lowercase](https://en.wikipedia.org/wiki/Letter_case) letters are treated as equivalent.
* Only the first 15 characters are significant.
* Keywords are reserved: you can’t use them as variable names.

Assuming function checkKeyword and relevant functions have been built.

**❷** MIDTERM TEST

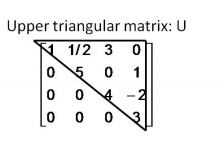
**COMPILER CONSTRUCTION**

(70 minutes- Open book- No Laptop)

(70 phút – được sử dụng tài liệu giấy, không được dùng các loại máy tính)

**Note:** The functions you are required to build are parts of project Parser used in practice exercise No 2. You must use the data structures declared in project Parser.

1/ An upper triangular matrix is a square matrix in which all entries below the main diagonal are zero. For example,



Write a KPL program to read elements in a matrix and check whether the matrix is upper triangular matrix or not. The program prints 1 if the matrix is upper triangular, otherwise it prints 0.

2/ Given the set of syntax rules:

**40) Constant ::= SB\_PLUS Constant2**

**41) Constant ::= SB\_MINUS Constant2**

**42) Constant ::= Constant2**

**43) Constant ::= TK\_CHAR**

**44) Constant2::= TK\_IDENT**

**45) Constant2::= TK\_NUMBER**

a. Prove that productions from 40 to 45 satisfy LL(1) condition

b. Write function void compileConstant(void) satisfying the syntax rules 40 to 43:

3/Write function void compileStatement (void) (and function void compileCallSt (void)if necessary) in case the procedure calls do not contain the keyword CALL, i.e. the syntax rule for procedure calls as follows:

CallSt ::= ProcedureIdent Arguments

Assuming function void compileArguments (void) has been built.