­BÁO CÁO THỰC HÀNH

CHƯƠNG TRÌNH DỊCH

LAB01

**Phạm Vân Anh**

**MSSV: 20214988**

1. **Hoàn thiện Scanner.c**

**State 2~3**

  case 2: {

    readChar();

    state = 0;

    return getToken();

  }

  case 3:

    {

      int count = 0;

      while (charCodes[currentChar] == CHAR\_LETTER || charCodes[currentChar] == CHAR\_DIGIT)

      {

        if(count < MAX\_IDENT\_LEN) {

            str[count] = currentChar;

            count++;

        }

        readChar();

      }

      str[count] = '\0';

      token = makeToken(TK\_IDENT, ln, cn);

      strcpy(token->string, str);

      state = 4;

      return getToken();

    }

**State 5~7**

  case 5:

    state = 0;

    return token;

  case 6:

    state = 0;

    return token;

  case 7:

    {

      int count = 0;

      char str[MAX\_IDENT\_LEN];

      token = makeToken(TK\_NUMBER, lineNo, colNo);

      while (charCodes[currentChar] == CHAR\_DIGIT)

      {

        if(count > 9) {

          error(ERR\_NUMBERTOOLONG, token->lineNo, token->colNo);

        } else {

            str[count] = currentChar;

            count++;

            readChar();

        }

      }

      str[count] = '\0';

      strcpy(token->string, str);

      token->value = atoi(token->string);

      state = 8;

      return getToken();

    }

**State 10~12**

  case 10:

    readChar();

    state = 0;

    return makeToken(SB\_MINUS, lineNo, colNo - 1);

  case 11:

    readChar();

    state = 0;

    return makeToken(SB\_TIMES, lineNo, colNo - 1);

  case 12:

    readChar();

    state = 0;

    return makeToken(SB\_SLASH, lineNo, colNo - 1);

**State 16~20**

  case 16:

    readChar();

    if (charCodes[currentChar] == CHAR\_EQ)

      state = 17;

    else

      state = 18;

    return getToken();

  case 17:

    ln = lineNo;

    cn = colNo;

    readChar();

    state = 0;

    return makeToken(SB\_GE, ln, cn-1);

  case 18:

    state = 0;

    return makeToken(SB\_GT, lineNo, colNo);

  case 19:

    ln = lineNo;

    cn = colNo;

    readChar();

    state = 0;

    return makeToken(SB\_EQ, ln, cn);

  case 20:

    readChar();

    if (charCodes[currentChar] == CHAR\_EQ)

      state = 21;

    else

      state = 22;

    return getToken();

**State 23~30**

  case 23:

    ln = lineNo;

    cn = colNo;

    readChar();

    state = 0;

    return makeToken(SB\_COMMA, ln, cn);

  case 24:

    readChar();

    ln = lineNo;

    cn = colNo;

    if (charCodes[currentChar] == CHAR\_RPAR)

      state = 25;

    else

      state = 26;

    return getToken();

  case 25:

    readChar();

    state = 0;

    return makeToken(SB\_RSEL, ln, cn-1);

  case 26:

    state = 0;

    return makeToken(SB\_PERIOD, ln, cn-1);

  case 27:

    ln = lineNo;

    cn = colNo;

    readChar();

    state = 0;

    return makeToken(SB\_SEMICOLON, ln, cn);

  case 28:

    readChar();

    if (charCodes[currentChar] == CHAR\_EQ)

      state = 29;

    else

      state = 30;

    return getToken();

  case 29:

    readChar();

    state = 0;

    return makeToken(SB\_ASSIGN, lineNo, colNo - 2);

  case 30:

    state = 0;

    return makeToken(SB\_COLON, lineNo, colNo - 1);

**State 36~39**

  case 36:

    readChar();

    state = 0;

    return makeToken(SB\_LSEL, lineNo, colNo - 2);

  case 37:

     readChar();

  while (currentChar != EOF) {

    if (charCodes[currentChar] == CHAR\_TIMES) {

      readChar();

      if (charCodes[currentChar] == CHAR\_RPAR) { // Phát hiện đóng chú thích \*)

        state = 39;

        break;

      }

    } else {

      readChar();

    }

  }

  if (currentChar == EOF) {

    error(ERR\_ENDOFCOMMENT, lineNo, colNo); // Báo lỗi nếu EOF trước khi kết thúc chú thích

  }

  return getToken();

  case 38:

    readChar();

    if(charCodes[currentChar] == CHAR\_RPAR) {

      state = 39;

    }

    return getToken();

  case 39:

  readChar();

  state = 0;

  return getToken();

1. **KẾT QUẢ**

* Biên dịch example1.kpl

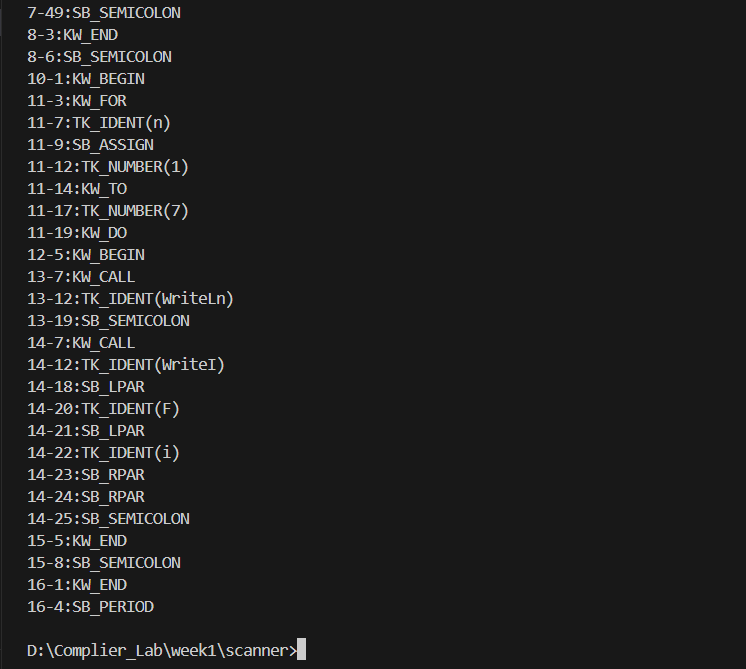
A screenshot of a computer

Description automatically generated

* Biên dịch example2.kpl

A screenshot of a computer program

Description automatically generated



* Biên dịch exampleA1.kpl, xử lí số nguyên lớn

A screenshot of a computer

Description automatically generated