Казанский (Приволжский) федеральный университет

Институт вычислительной математики и информационных технологий

Отчёт по дисциплине «Пакеты прикладных программ»

студента гр. 09 – 712 Гимадутдинова Р.М.

Работу выполнил:

Студент 09-712 группы

Гимадутдинов Рустем Маратович

Работу проверил:

Доцент кафедры теоретической кибернетики

Гусенков Александр Михайлович

Казань 2020

Оглавление

[Индивидуальное задание 3](#_Toc59019603)

[Входной файл для генератора lex 5](#_Toc59019604)

[Входной файл для генератора yacc 8](#_Toc59019605)

[symbols.h - заголовочный файл модуля для работы с переменными 13](#_Toc59019606)

[symbols.c – модуль для работы с переменными 15](#_Toc59019607)

[Тестовый файл 17](#_Toc59019608)

[Результат работы на тестовом файле 20](#_Toc59019609)

# Индивидуальное задание

**Команда SUM**

Суммирует все или только заданные числовые поля в таблице, выбранной в данный момент.

**Синтаксис:**

*SUM [eExpressionList]*

*[Scope] [FOR lExpression1] [WHILE lExpression2]*

*[TO MemVarNameList | TO ARRAY ArrayName]*

*[NOOPTIMIZE]*

**Параметры:**

*eExpressionList*

Задает одно или несколько полей или выражений полей, которые требуется п осуммировать. Если список выражений полей опущен, суммируются все числовые поля.

*Scope*

Задает диапазон записей, включаемых в общую сумму. Диапазон определяется сле ующими предложениями: ALL, NEXT nRecords, RECORD nRecordNumber и REST. Подробнее о предложения диапазона см. тему Предложения диапазона или главу 2 "Обзор языка программирования" Руководства разработчика.

По умолчанию областью действия команды SUM являются все записи (предложение ALL).

*FOR lExpression1*

Задает суммирование только тех записей, для которых логическое условие lExpression1 имеет значение "истина" (.T.). Применение предложения FOR позволяет выполнять условное суммирование записей, отсеивая ненужные.

Технология Rushmore оптимизирует команду SUM ... FOR, если lExpression1 является оптимизируемым выражением. В целях получения наивысшей производитель ости используйте в предложении FOR оптимизируемое выражение. Подробнее об оптимизации по технологии Rushmore см. темы SET OPTIMIZE и Основы технологии Rushmore или главу 17 " Оптимизация приложений" Руководства разработчика.

*WHILE lExpression2*

Задает условие, в соответствии с которым записи из текущей таблицы включаются в общую сумму, пока логическое выражение lExpression2 остается истинным (.T.).

*TO MemVarNameList*

Сохраняет каждую сумму в переменной памяти. Если в списке MemVarNameList указана несуществующая переменная памяти, Visual FoxPro автоматически создает ее. Разделяйте имена переменных в списке запятыми.

*TO ARRAY ArrayName*

Сохраняет суммы в массиве переменных памяти. Если в команде SUM задан есуществующий массив, Visual FoxPro автоматически создает его. Если массив существует, но слишком мал для того, чтобы вместить все полученные суммы, его размер автоматически увеличивается надлежащим образом.

*NOOPTIMIZE*

Блокирует оптимизацию команды SUM по технологии Rushmore. Подробнее см. темы SET OPTIMIZE и Основы технологии Rushmore или главу 17 " Оптимизация приложений" Руководства разработчика.

Будем рассматривать язык в котором допустимы следующие два вида команд:

* Команда SUM
* Оператор присваивания

*Пример:*

SUM W\*(Y + per), salary WHILE Y == 0;

rTwo = W + 2;

# Входной файл для генератора lex

%option yylineno

%{

#include "y.tab.h"

#include "symbols.h"

int pos = 0;

int err\_cnt = 0;

%}

%%

\xD ;

[Ss][Uu][Mm] { pos += yyleng; printf("SUM\n"); return ySUM; }

[Ff][Oo][Rr] { pos += yyleng; printf("FOR\n"); return yFOR; }

[Ww][Hh][Ii][Ll][Ee] { pos += yyleng; printf("WHILE\n"); return yWHILE; }

[Tt][Oo] { pos += yyleng; printf("TO\n"); return yTO; }

[Aa][Rr][Rr][Aa][Yy] { pos += yyleng; printf("ARRAY\n"); return yARRAY; }

[Nn][Oo][Oo][Pp][Tt][Ii][Mm][Ii][Zz][Ee] { pos += yyleng; printf("NOOPTIMIZE\n"); return yNOOPTIMIZE; }

[Aa][Ll][Ll] { pos += yyleng; printf("ALL\n"); return yALL; }

[Nn][Ee][Xx][Tt] { pos += yyleng; printf("NEXT\n"); return yNEXT; }

[Rr][Ee][Cc][Oo][Rr][Dd] { pos += yyleng; printf("RECORD\n"); return yRECORD; }

[Rr][Ee][Ss][Tt] { pos += yyleng; printf("REST\n"); return yREST; }

"," { pos += yyleng; printf("COMMA\n"); return yCOMMA; }

"+" { pos += yyleng; printf("PLUS\n"); return yPLUS; }

"-" { pos += yyleng; printf("MINUS\n"); return yMINUS; }

"\*" { pos += yyleng; printf("MULTIPLY\n"); return yMULTIPLY; }

"/" { pos += yyleng; printf("DIVIDE\n"); return yDIVIDE; }

"<=" { pos += yyleng; printf("LEQ\n"); return yLEQ; }

">=" { pos += yyleng; printf("GEQ\n"); return yGEQ; }

"<>"|"#"|"!=" { pos += yyleng; printf("NEQ\n"); return yNEQ; }

"==" { pos += yyleng; printf("STR\_EQ\n"); return ySTR\_EQ; }

"=" { pos += yyleng; printf("EQUAL\n"); return yEQUAL; }

"<" { pos += yyleng; printf("LESS\n"); return yLESS; }

">" { pos += yyleng; printf("GREATER\n"); return yGREATER; }

[Aa][Nn][Dd] { pos += yyleng; printf("AND\n"); return yAND; }

[Oo][Rr] { pos += yyleng; printf("OR\n"); return yOR; }

[Nn][Oo][Tt] { pos += yyleng; printf("NOT\n"); return yNOT; }

"(" { pos += yyleng; printf("OP\n"); return yOP; }

")" { pos += yyleng; printf("CP\n"); return yCP; }

[a-zA-Z][a-zA-Z0-9]\* { pos += yyleng; printf("IDENTIFIER = %s\n", yytext); yylval.str = strdup(yytext); return yIDENTIFIER; }

0|([1-9][0-9]\*) {

yylval.num = yytext[0] - '0';

int i;

for (i = 1; i < yyleng; i++) { yylval.num = yylval.num\*10 + (yytext[i] - '0'); }

printf("NUMBER = %d\n", yylval.num);

return yNUMBER;

}

; { printf("SEMICOLON\n"); return ySEMICOLON; }

\n { pos = 0; }

" " { pos++; }

[ \t]+ ;

. { pos++; err\_cnt++; printf("at line %d, position %d : lexical error: unknown character '%c'\n", yylineno, pos, yytext[0]); }

%%

# Входной файл для генератора yacc

%union

{

int num;

char \*str;

}

%token ySUM yFOR yWHILE yTO yARRAY yNOOPTIMIZE yALL yNEXT yRECORD yREST

%token yCOMMA yPLUS yMINUS yMULTIPLY yDIVIDE yLEQ yGEQ yNEQ ySTR\_EQ yEQUAL yLESS yGREATER

%token <str> yIDENTIFIER

%token <num> yNUMBER

%token yAND yOR yNOT yOP yCP ySEMICOLON yANY

%type <num> exp term

%left yOR

%left yAND

%left yNOT

%left yLESS yGREATER yLEQ yGEQ yNEQ ySTR\_EQ yEQUAL

%left yPLUS yMINUS

%left yMULTIPLY yDIVIDE

%precedence UMINUS

%{

#include <stdio.h>

#include <stdlib.h>

#include "symbols.h"

extern int pos;

extern int yylineno;

extern int err\_cnt;

extern int yylex(void);

void yyerror(char \*);

void runtime\_error(char \*str);

symbol\_block \*symtab = NULL;

%}

%%

list:

| list stat ySEMICOLON

| list stat error { yyerror("semicolon is missing"); }

| list error ySEMICOLON { yyerror("wrong statement"); }

stat: ySUM exp\_list scope for\_exp while\_exp to\_exp optimize\_exp

| yIDENTIFIER yEQUAL exp {

/\*printf("%s = %d\n", $1, $3);\*/

printf("expression value = %d\n", $3);

symbol\_assign(&symtab, $1, $3);

}

| yIDENTIFIER error exp { yyerror("assignment expected"); }

;

exp\_list:

| exp { printf("expression value = %d\n", $1); }

| exp\_list yCOMMA exp { printf("expression value = %d\n", $3); }

;

lexp: exp yLESS exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp yGREATER exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp yLEQ exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp yGEQ exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp yNEQ exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp ySTR\_EQ exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp yEQUAL exp { printf("left expression value = %d\nright expression value = %d\n", $1, $3); }

| exp error exp { yyerror("comparison sign is missing"); }

| lexp yAND lexp

| lexp yOR lexp

| yNOT lexp

| yOP lexp yCP

;

exp:

term

| exp yPLUS exp { $$ = $1 + $3; }

| exp yMINUS exp { $$ = $1 - $3; }

| exp yMULTIPLY exp { $$ = $1 \* $3; }

| exp yDIVIDE exp {

if ($3 == 0) {

runtime\_error("division by zero.");

} else {

$$ = $1 / $3;

}

}

| yMINUS exp %prec UMINUS { $$ = -$2; }

| yOP exp yCP { $$ = $2; }

;

term:

yNUMBER { $$ = $1; }

| yIDENTIFIER {

int dst;

if (symbol\_get\_value(&symtab, &dst, $1) == 1) {

$$ = dst;

} else {

char msg[80];

sprintf(msg, "unknown identifier %s", $1);

runtime\_error(msg);

$$ = 9999999;

}

}

;

scope:

| yALL

| yNEXT exp { printf("expression value = %d\n", $2); }

| yNEXT error { yyerror("expected an expression after NEXT"); }

| yRECORD exp { printf("expression value = %d\n", $2); }

| yRECORD error { yyerror("expected an expression after RECORD"); }

| yREST

;

for\_exp:

| yFOR lexp

;

while\_exp:

| yWHILE lexp

;

to\_exp:

| yTO var\_list

| yTO yARRAY yIDENTIFIER

;

var\_list: yIDENTIFIER

| var\_list yCOMMA yIDENTIFIER

;

optimize\_exp:

| yNOOPTIMIZE

;

%%

int main(int argc, char \*\*argv) {

yyparse();

if (err\_cnt == 0) {

printf("Parsed successfully.\n");

} else {

printf("Parsed with %d errors.\n", err\_cnt);

}

}

void yyerror(char \*str) {

if (strcmp(str, "syntax error")) { printf("at line %d, position %d : syntax error: %s\n", yylineno, pos, str); err\_cnt++; }

}

void runtime\_error(char \*str) {

printf("at line %d, position %d : runtime error: %s\n", yylineno, pos, str); err\_cnt++;

}

## symbols.h - заголовочный файл модуля для работы с переменными

В целях улучшения читаемости кода и структуры проекта, работа с перемнными была вынесена в отдельный модуль symbols.

#ifndef SYMBOLS\_H

#define SYMBOLS\_H

#include <string.h>

#include <stdlib.h>

#define BLOCK\_SIZE 15

#define NAME\_LEN 30

typedef struct symbol {

char name[NAME\_LEN]; // symbol name

int val; // symbol value

int hasVal; // 0 if symbol has notany value, 1 if has

} symbol;

typedef struct symbol\_block {

symbol \*symbols; // pointer to symbol array of BLOCK\_SIZE size

int cur\_index; // current index put new element to

struct symbol\_block \*next; // pointer to the next block

} symbol\_block;

/\* Read pointer to symbol to dst.

Return value:

-1 if symbols not found

0 if symbol found but hasVal is 0

1 if symbol found and hasVal is 1

Note: this function can be used to just check if some symbol exist or not\*/

int symbol\_get(symbol\_block \*\*root, symbol \*\*dst, char \*name);

/\* Read symbol value to dst.

Return value:

-1 if symbols not found

0 if symbol found but hasVal is 0

1 if symbol found and hasVal is 1

Note: this function can be used to just check if some symbol exist or not\*/

int symbol\_get\_value(symbol\_block \*\*root, int \*dst, char \*name);

/\* Add symbol without assigning it any value

Note: if symbol already exists, function won't add a new one \*/

void symbol\_add(symbol\_block \*\*root, char \*name);

/\* Add symbol and assign given value to it

Note: if symbol already exists, function will change its value,

if doesnt, function will create a new one with given value \*/

void symbol\_assign(symbol\_block \*\*root, char \*name, int value);

/\* If root block is filled up, then alloc new and insert it before root \*/

void enlarge\_if\_need(symbol\_block \*\*root);

/\*

Example usage:

symbol\_block \*root;

...

int x;

symbol\_get\_value(&root, &x, "foobar");

symbol \*sym;

symbol\_get(&root, &sym, "foo");

\*/

#endif // SYMBOLS\_H

## symbols.c – модуль для работы с переменными

Ниже приведена реализация модуля symbols.

#include "symbols.h"

int symbol\_get(symbol\_block \*\*root, symbol \*\*dst, char \*name) {

symbol\_block \*cur\_block = \*root; int i;

while (cur\_block != NULL) {

for (i = 0; i < cur\_block->cur\_index; i++) {

if (strcmp(cur\_block->symbols[i].name, name) == 0) {

if (cur\_block->symbols[i].hasVal == 1) {

\*dst = &(cur\_block->symbols[i]);

return 1;

}

else { return 0; }

}

}

cur\_block = cur\_block->next;

}

return -1;

}

int symbol\_get\_value(symbol\_block \*\*root, int \*dst, char \*name) {

symbol \*sym;

int retval = symbol\_get(root, &sym, name);

if (retval == 1) {

\*dst = sym->val;

}

return retval;

}

void enlarge\_if\_need(symbol\_block \*\*root)

{

if (\*root == NULL || (\*root)->cur\_index >= BLOCK\_SIZE)

{

symbol\_block \*sb = (symbol\_block\*)malloc(sizeof(symbol\_block));

sb->symbols = (symbol\*)malloc(BLOCK\_SIZE\*sizeof(symbol));

sb->cur\_index = 0;

sb->next = \*root;

\*root = sb;

}

}

void symbol\_add(symbol\_block \*\*root, char \*name)

{

int tmp; if (symbol\_get\_value(root, &tmp, name) != -1) { return; }

enlarge\_if\_need(root);

int ind = (\*root)->cur\_index;

strcpy((\*root)->symbols[ind].name, name);

(\*root)->symbols[ind].hasVal = 0;

(\*root)->cur\_index++;

}

void symbol\_assign(symbol\_block \*\*root, char \*name, int value)

{

symbol \*sym;

if (symbol\_get(root, &sym, name) != -1) {

sym->val = value;

sym->hasVal = 1;

return;

}

enlarge\_if\_need(root);

int ind = (\*root)->cur\_index;

strcpy((\*root)->symbols[ind].name, name);

(\*root)->symbols[ind].hasVal = 1;

(\*root)->symbols[ind].val = value;

(\*root)->cur\_index++;

}

# Тестовый файл

Замечание: строки пронумерованы.

1. abc = 12;

2. abc = 13;

3. vbf = 234@@'@;

4. SuM abc;

5. Y = 8 - 6

6. dengi = 123 - 7 + abc;

7. H H;

8. per = dengi - 9;

9. pole3 = Y\*Y;

10. yyy =pole3 / (5 -3 -2);

11. W = Y\*Y\*Y;

12. rOne = W;

13. rTwo = W + 2;

14. xyz = (2\*(2 + abc + Y) + 88 - abc)\*(7 - 5);

15. gdgfhg = -9 \* -7;

16. SUM xyz + 7 NEXT kk FOR dengi - 1 <= 44 WHILE dengi = 13 - Y;

17. SuM per/6 RECORD 13 FOR pole3 <> 1 AND dengi + 2 > 0;

18. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

19. SUM W\*(Y - per), abc WHILE Y != 0;

20. SUM xyz + 7 NEXT FOR dengi - 1 >= 0 WHILE prof = 177 - rTwo;

21. SuM per/6 RECORD 13 FOR pole3 <> 1 OR dengi + 2 > 0;

22. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND abc = 3 TO rOne, rTwo, W;

23. SUM W\*(Y - 2 - pergef), salary WHILE Y == 0;

24. abc = 127/0;

25. Y = 8 - 6;

26. dengi = 123;

27. per = dengi - 9 + privetmir;

28. pole3 = Y\*Y;

29. W = Y\*Y\*Y\*Y;

30. SuM per/6 RECORD 13 FOR pole3 <> 1 OR NOT dengi + 7 > 0;

31. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

32. SUM W\*(Y + per), salary WHILE Y == 0;

33. rOne = W;

34. rTwo = W + 2;

35. xyz = 2\*(2 + abc + 777 + Y);

36. abc = 199;

37. SUM abc + 7 NEXT 56 FOR dengi - 1 <= 23 WHILE prof = 13 - Y;

38. SuM per/6 RECORD 13 FOR pole3 <> 1 AND dengi + 2 > 0;

39. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

40. SUM W\*(Y - per), salary WHILE rTwo != 0;

41. SUM xyz + 7 NEXT 56 FOR dengi - 1 <= 23 WHILE prof = 13 - Y;

42. SuM per/6 RECORD 13 FOR pole3 <> 1 AND dengi + 2 > 0;

43. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

44. SUM W\*(Y - per\*2), salary WHILE Y != 0;

45. SuM per/6 RECORD 13 FOR pole3 <> 1 OR NOT dengi + 7 > 0;

46. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

47. SUM 123+88-99, salary WHILE Y == 0;

48. rOne = W;

49. rTwo = W + 2;

50. xyz = 2\*(2 + abc + 777 + Y);

51. abc = 12;

52. SUM abc/7 NEXT 56 FOR dengi - 1 <= 23 WHILE prof = 13 - Y;

53. SuM per/6 RECORD 13 FOR pole3 <> 1 AND dengi + 2 > 0;

54. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

55. SUM W\*(Y - per), salary WHILE rTwo != 0;

56. SUM xyz + 7 NEXT 56 FOR dengi - 1 <= 23 WHILE prof = 13 - Y;

57. SuM per/6 RECORD 13 FOR pole3 <> 1 AND dengi + 2 > 0;

58. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

59. SUM W\*(Y - per\*2), salary WHILE Y != 0;

60. Y = 8 - 6;

61. dengi = 123;

62. per = dengi - 9;

63. pole3 = Y\*Y;

64. W = Y\*Y\*Y\*Y;

65. gdgfhg = -9 \* -7;

66. SUM xyz + 7 NEXT kk FOR dengi - 1 <= 44 WHILE dengi = 13 - Y;

67. SuM per/6 RECORD 13 FOR pole3 1 AND dengi + 2 > 0;

68. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

69. SUM W\*(Y - per), abc WHILE Y != 0;

70. SUM xyz + 7 NEXT 56 FOR dengi - 1 >= 0 WHILE prof = 177 - rTwo;

71. SuM per/6 RECORD 13 FOR pole3 <> 1 OR dengi + 2 > 0;

72. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND abc = 3 TO rOne, rTwo, W;

73. SUM W\*(Y - 2 - per)\*3\*(7 - 5), salary WHILE Y == 0;

74. abc = 12;

75. Y = 8 - 6;

76. dengi = 123;

77. per = dengi - 9;

78. pole3 = Y\*Y;

79. W = Y\*Y\*Y\*Y;

80. SuM per/6 RECORD 13 FOR pole3 <> 1 OR NOT dengi + 7 > 0;

81. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

82. SUM W\*(Y + per), salary WHILE Y == 0;

83. rOne = W;

84. rTwo = W + 2;

85. xyz = 2\*(2 + abc + 777 + Y);

86. abc = 12;

87. SUM abc + 7 NEXT 56 FOR dengi - 1 <= 23 WHILE prof = 13 - Y;

88. SuM per/6 RECORD 13 FOR pole3 <> 1 AND dengi + 2 > 0;

89. sum Y\*Y, W\*3, W\*2 FOR W <= 2 AND dengi = 3 TO rOne, rTwo, W;

90. SUM W\*(Y - per), salary WHILE rTwo != 0;

91. SUM xyz + 7 NEXT 56 FOR dengi - 1 <= 23 WHILE prof = 13 - Y;

# Результат работы на тестовом файле

Запустим получившуюся программу на тестовом файле, получим вывод:

IDENTIFIER = abc

EQUAL

NUMBER = 12

SEMICOLON

expression value = 12

IDENTIFIER = abc

EQUAL

NUMBER = 13

SEMICOLON

expression value = 13

IDENTIFIER = vbf

EQUAL

NUMBER = 234

at line 3, position 7 : lexical error: unknown character '@'

at line 3, position 8 : lexical error: unknown character '@'

at line 3, position 9 : lexical error: unknown character '''

at line 3, position 10 : lexical error: unknown character '@'

SEMICOLON

expression value = 234

SUM

IDENTIFIER = abc

SEMICOLON

expression value = 13

IDENTIFIER = Y

EQUAL

NUMBER = 8

MINUS

NUMBER = 6

IDENTIFIER = dengi

expression value = 2

at line 6, position 5 : syntax error: semicolon is missing

EQUAL

NUMBER = 123

MINUS

NUMBER = 7

PLUS

IDENTIFIER = abc

SEMICOLON

expression value = 129

IDENTIFIER = H

IDENTIFIER = H

at line 7, position 3 : runtime error: unknown identifier H

SEMICOLON

at line 7, position 3 : syntax error: assignment expected

IDENTIFIER = per

EQUAL

IDENTIFIER = dengi

MINUS

NUMBER = 9

SEMICOLON

expression value = 120

IDENTIFIER = pole3

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 4

IDENTIFIER = yyy

EQUAL

IDENTIFIER = pole3

DIVIDE

OP

NUMBER = 5

MINUS

NUMBER = 3

MINUS

NUMBER = 2

CP

at line 10, position 19 : runtime error: division by zero.

SEMICOLON

expression value = 4

IDENTIFIER = W

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 8

IDENTIFIER = rOne

EQUAL

IDENTIFIER = W

SEMICOLON

expression value = 8

IDENTIFIER = rTwo

EQUAL

IDENTIFIER = W

PLUS

NUMBER = 2

SEMICOLON

expression value = 10

IDENTIFIER = xyz

EQUAL

OP

NUMBER = 2

MULTIPLY

OP

NUMBER = 2

PLUS

IDENTIFIER = abc

PLUS

IDENTIFIER = Y

CP

PLUS

NUMBER = 88

MINUS

IDENTIFIER = abc

CP

MULTIPLY

OP

NUMBER = 7

MINUS

NUMBER = 5

CP

SEMICOLON

expression value = 218

IDENTIFIER = gdgfhg

EQUAL

MINUS

NUMBER = 9

MULTIPLY

MINUS

NUMBER = 7

SEMICOLON

expression value = 63

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 225

IDENTIFIER = kk

at line 16, position 18 : runtime error: unknown identifier kk

FOR

expression value = 9999999

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 44

WHILE

left expression value = 128

right expression value = 44

IDENTIFIER = dengi

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 129

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 20

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

AND

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 131

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 24

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 16

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 8

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 129

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

CP

COMMA

expression value = -944

IDENTIFIER = abc

WHILE

expression value = 13

IDENTIFIER = Y

NEQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 225

FOR

at line 20, position 18 : syntax error: expected an expression after NEXT

IDENTIFIER = dengi

MINUS

NUMBER = 1

GEQ

NUMBER = 0

WHILE

left expression value = 128

right expression value = 0

IDENTIFIER = prof

at line 20, position 42 : runtime error: unknown identifier prof

EQUAL

NUMBER = 177

MINUS

IDENTIFIER = rTwo

SEMICOLON

left expression value = 9999999

right expression value = 167

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 20

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

OR

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 131

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 24

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 16

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 8

right expression value = 2

IDENTIFIER = abc

EQUAL

NUMBER = 3

TO

left expression value = 13

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

NUMBER = 2

MINUS

IDENTIFIER = pergef

at line 23, position 20 : runtime error: unknown identifier pergef

CP

COMMA

expression value = -79999992

IDENTIFIER = salary

at line 23, position 29 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

STR\_EQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

IDENTIFIER = abc

EQUAL

NUMBER = 127

DIVIDE

NUMBER = 0

at line 24, position 7 : runtime error: division by zero.

SEMICOLON

expression value = 127

IDENTIFIER = Y

EQUAL

NUMBER = 8

MINUS

NUMBER = 6

SEMICOLON

expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 123

SEMICOLON

expression value = 123

IDENTIFIER = per

EQUAL

IDENTIFIER = dengi

MINUS

NUMBER = 9

PLUS

IDENTIFIER = privetmir

at line 27, position 26 : runtime error: unknown identifier privetmir

SEMICOLON

expression value = 10000113

IDENTIFIER = pole3

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 4

IDENTIFIER = W

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 16

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 1666685

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

OR

left expression value = 4

right expression value = 1

NOT

IDENTIFIER = dengi

PLUS

NUMBER = 7

GREATER

NUMBER = 0

SEMICOLON

left expression value = 130

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

PLUS

IDENTIFIER = per

CP

COMMA

expression value = 160001840

IDENTIFIER = salary

at line 32, position 23 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

STR\_EQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

IDENTIFIER = rOne

EQUAL

IDENTIFIER = W

SEMICOLON

expression value = 16

IDENTIFIER = rTwo

EQUAL

IDENTIFIER = W

PLUS

NUMBER = 2

SEMICOLON

expression value = 18

IDENTIFIER = xyz

EQUAL

NUMBER = 2

MULTIPLY

OP

NUMBER = 2

PLUS

IDENTIFIER = abc

PLUS

NUMBER = 777

PLUS

IDENTIFIER = Y

CP

SEMICOLON

expression value = 1816

IDENTIFIER = abc

EQUAL

NUMBER = 199

SEMICOLON

expression value = 199

SUM

IDENTIFIER = abc

PLUS

NUMBER = 7

NEXT

expression value = 206

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 23

WHILE

left expression value = 122

right expression value = 23

IDENTIFIER = prof

at line 37, position 44 : runtime error: unknown identifier prof

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 9999999

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 1666685

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

AND

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

CP

COMMA

expression value = -160001776

IDENTIFIER = salary

at line 40, position 23 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = rTwo

NEQ

NUMBER = 0

SEMICOLON

left expression value = 18

right expression value = 0

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 1823

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 23

WHILE

left expression value = 122

right expression value = 23

IDENTIFIER = prof

at line 41, position 44 : runtime error: unknown identifier prof

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 9999999

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 1666685

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

AND

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

MULTIPLY

NUMBER = 2

CP

COMMA

expression value = -320003584

IDENTIFIER = salary

at line 44, position 24 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

NEQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 1666685

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

OR

left expression value = 4

right expression value = 1

NOT

IDENTIFIER = dengi

PLUS

NUMBER = 7

GREATER

NUMBER = 0

SEMICOLON

left expression value = 130

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

NUMBER = 123

PLUS

NUMBER = 88

MINUS

NUMBER = 99

COMMA

expression value = 112

IDENTIFIER = salary

at line 47, position 14 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

STR\_EQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

IDENTIFIER = rOne

EQUAL

IDENTIFIER = W

SEMICOLON

expression value = 16

IDENTIFIER = rTwo

EQUAL

IDENTIFIER = W

PLUS

NUMBER = 2

SEMICOLON

expression value = 18

IDENTIFIER = xyz

EQUAL

NUMBER = 2

MULTIPLY

OP

NUMBER = 2

PLUS

IDENTIFIER = abc

PLUS

NUMBER = 777

PLUS

IDENTIFIER = Y

CP

SEMICOLON

expression value = 1960

IDENTIFIER = abc

EQUAL

NUMBER = 12

SEMICOLON

expression value = 12

SUM

IDENTIFIER = abc

DIVIDE

NUMBER = 7

NEXT

expression value = 1

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 23

WHILE

left expression value = 122

right expression value = 23

IDENTIFIER = prof

at line 52, position 42 : runtime error: unknown identifier prof

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 9999999

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 1666685

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

AND

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

CP

COMMA

expression value = -160001776

IDENTIFIER = salary

at line 55, position 23 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = rTwo

NEQ

NUMBER = 0

SEMICOLON

left expression value = 18

right expression value = 0

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 1967

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 23

WHILE

left expression value = 122

right expression value = 23

IDENTIFIER = prof

at line 56, position 44 : runtime error: unknown identifier prof

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 9999999

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 1666685

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

AND

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

MULTIPLY

NUMBER = 2

CP

COMMA

expression value = -320003584

IDENTIFIER = salary

at line 59, position 24 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

NEQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

IDENTIFIER = Y

EQUAL

NUMBER = 8

MINUS

NUMBER = 6

SEMICOLON

expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 123

SEMICOLON

expression value = 123

IDENTIFIER = per

EQUAL

IDENTIFIER = dengi

MINUS

NUMBER = 9

SEMICOLON

expression value = 114

IDENTIFIER = pole3

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 4

IDENTIFIER = W

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 16

IDENTIFIER = gdgfhg

EQUAL

MINUS

NUMBER = 9

MULTIPLY

MINUS

NUMBER = 7

SEMICOLON

expression value = 63

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 1967

IDENTIFIER = kk

at line 66, position 18 : runtime error: unknown identifier kk

FOR

expression value = 9999999

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 44

WHILE

left expression value = 122

right expression value = 44

IDENTIFIER = dengi

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 123

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 19

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NUMBER = 1

AND

at line 67, position 30 : syntax error: comparison sign is missing

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

CP

COMMA

expression value = -1792

IDENTIFIER = abc

WHILE

expression value = 12

IDENTIFIER = Y

NEQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 1967

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

GEQ

NUMBER = 0

WHILE

left expression value = 122

right expression value = 0

IDENTIFIER = prof

at line 70, position 44 : runtime error: unknown identifier prof

EQUAL

NUMBER = 177

MINUS

IDENTIFIER = rTwo

SEMICOLON

left expression value = 9999999

right expression value = 159

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 19

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

OR

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = abc

EQUAL

NUMBER = 3

TO

left expression value = 12

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

NUMBER = 2

MINUS

IDENTIFIER = per

CP

MULTIPLY

NUMBER = 3

MULTIPLY

OP

NUMBER = 7

MINUS

NUMBER = 5

CP

COMMA

expression value = -10944

IDENTIFIER = salary

at line 73, position 33 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

STR\_EQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

IDENTIFIER = abc

EQUAL

NUMBER = 12

SEMICOLON

expression value = 12

IDENTIFIER = Y

EQUAL

NUMBER = 8

MINUS

NUMBER = 6

SEMICOLON

expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 123

SEMICOLON

expression value = 123

IDENTIFIER = per

EQUAL

IDENTIFIER = dengi

MINUS

NUMBER = 9

SEMICOLON

expression value = 114

IDENTIFIER = pole3

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 4

IDENTIFIER = W

EQUAL

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

SEMICOLON

expression value = 16

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 19

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

OR

left expression value = 4

right expression value = 1

NOT

IDENTIFIER = dengi

PLUS

NUMBER = 7

GREATER

NUMBER = 0

SEMICOLON

left expression value = 130

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

PLUS

IDENTIFIER = per

CP

COMMA

expression value = 1856

IDENTIFIER = salary

at line 82, position 23 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = Y

STR\_EQ

NUMBER = 0

SEMICOLON

left expression value = 2

right expression value = 0

IDENTIFIER = rOne

EQUAL

IDENTIFIER = W

SEMICOLON

expression value = 16

IDENTIFIER = rTwo

EQUAL

IDENTIFIER = W

PLUS

NUMBER = 2

SEMICOLON

expression value = 18

IDENTIFIER = xyz

EQUAL

NUMBER = 2

MULTIPLY

OP

NUMBER = 2

PLUS

IDENTIFIER = abc

PLUS

NUMBER = 777

PLUS

IDENTIFIER = Y

CP

SEMICOLON

expression value = 1586

IDENTIFIER = abc

EQUAL

NUMBER = 12

SEMICOLON

expression value = 12

SUM

IDENTIFIER = abc

PLUS

NUMBER = 7

NEXT

expression value = 19

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 23

WHILE

left expression value = 122

right expression value = 23

IDENTIFIER = prof

at line 87, position 44 : runtime error: unknown identifier prof

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 9999999

right expression value = 11

SUM

IDENTIFIER = per

DIVIDE

NUMBER = 6

RECORD

expression value = 19

NUMBER = 13

FOR

expression value = 13

IDENTIFIER = pole3

NEQ

NUMBER = 1

AND

left expression value = 4

right expression value = 1

IDENTIFIER = dengi

PLUS

NUMBER = 2

GREATER

NUMBER = 0

SEMICOLON

left expression value = 125

right expression value = 0

SUM

IDENTIFIER = Y

MULTIPLY

IDENTIFIER = Y

COMMA

expression value = 4

IDENTIFIER = W

MULTIPLY

NUMBER = 3

COMMA

expression value = 48

IDENTIFIER = W

MULTIPLY

NUMBER = 2

FOR

expression value = 32

IDENTIFIER = W

LEQ

NUMBER = 2

AND

left expression value = 16

right expression value = 2

IDENTIFIER = dengi

EQUAL

NUMBER = 3

TO

left expression value = 123

right expression value = 3

IDENTIFIER = rOne

COMMA

IDENTIFIER = rTwo

COMMA

IDENTIFIER = W

SEMICOLON

SUM

IDENTIFIER = W

MULTIPLY

OP

IDENTIFIER = Y

MINUS

IDENTIFIER = per

CP

COMMA

expression value = -1792

IDENTIFIER = salary

at line 90, position 23 : runtime error: unknown identifier salary

WHILE

expression value = 9999999

IDENTIFIER = rTwo

NEQ

NUMBER = 0

SEMICOLON

left expression value = 18

right expression value = 0

SUM

IDENTIFIER = xyz

PLUS

NUMBER = 7

NEXT

expression value = 1593

NUMBER = 56

FOR

expression value = 56

IDENTIFIER = dengi

MINUS

NUMBER = 1

LEQ

NUMBER = 23

WHILE

left expression value = 122

right expression value = 23

IDENTIFIER = prof

at line 91, position 44 : runtime error: unknown identifier prof

EQUAL

NUMBER = 13

MINUS

IDENTIFIER = Y

SEMICOLON

left expression value = 9999999

right expression value = 11

Parsed with 33 errors.