

Лабораторная работа №3

Программирование автомата «Транслятор»

Запрограммировать автомат в соответствии с данным описанием.

Требования к программе.

1. Программа должна быть написана на C++ Builder.
2. Должен быть введёна иерархия классов:
tr = "translator.txt";
opd = "operand.txt";
tm = "term.txt";
ex = "expression.txt";
opr = "operator.txt";
st = "stack.txt";
3. Рекомендуется в качестве main.cpp использовать прилагаемую программу.
4. Разработанная программа должна быть протестирована сравнением с прилагаемой программой Translator.exe.

Автомат «Транслятор» (верхний уровень)

Таблица переходов автомата «Транслятор»:

	z0	z1	z2
Не спец. симв.	0	0	-
;	1	-	-
#	-	2	-

Действия по состояниям:

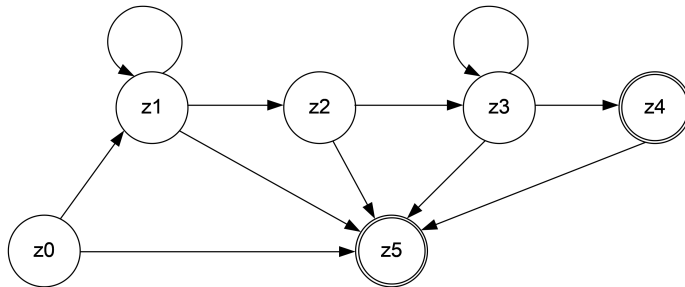
- z0: подаём поступивший на вход символ автомату «Оператор»;
z1: подаём поступивший на вход символ автомату «Оператор», сброс в 0 автомата «Оператор»;
z2: печать результата.

Автомат «Оператор»

Таблица переходов автомата «Оператор»:

	z0	z1	z2	z3	z4	z5
буква	1	1	3	3	-	-
цифра	5	1	3	3	-	-
(5	5	3	3	-	-
)	5	5	5	3	-	-
=	5	2	5	5	-	-
;	5	5	5	4	-	-
+ - * / .	5	5	5	3	-	-
запр.	5	5	5	5	-	-

Граф переходов автомата «Оператор»



Действия по состояниям:

z0: начальное состояние

z1: добавляем символ, поступивший на вход, в к имени переменной

z2: вталкиваем символ '=' поступивший на вход в стек

z3: передаем поступивший на вход символ в автомат «Стек1» (символы после '=')

z4: финальное состояние считывания новой переменной, заносим её имя в словарь

z5: состояние ошибки

Автоматы «Стек1» и «Стек2»

Таблица переходов автомата «Стек»:

Стек1	Стек2	z0	z1	z2
другие	другие	1	1	1
) ;	(=	2	2	2

Действия по состояниям «Стек1»:

z0: сброс

z1: добавляем символ, поступивший на вход, в «Стек1»

z2: переписываем «Стек1» в автомат «Стек2»

Действия по состояниям «Стек2»:

z0: сброс

z1: добавляем символ, поступивший на вход, в «Стек2»

z2: переписываем «Стек2» в автомат «Выражение», затем результат работы автомата «Выражение», преобразуем в строку и посимвольно подаем на вход «Стек1»

Автомат «Выражение»

Семантическая диаграмма:

ВЫРАЖЕНИЕ(a)::

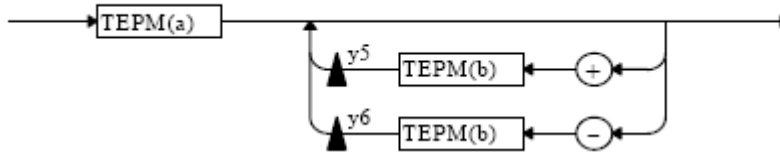
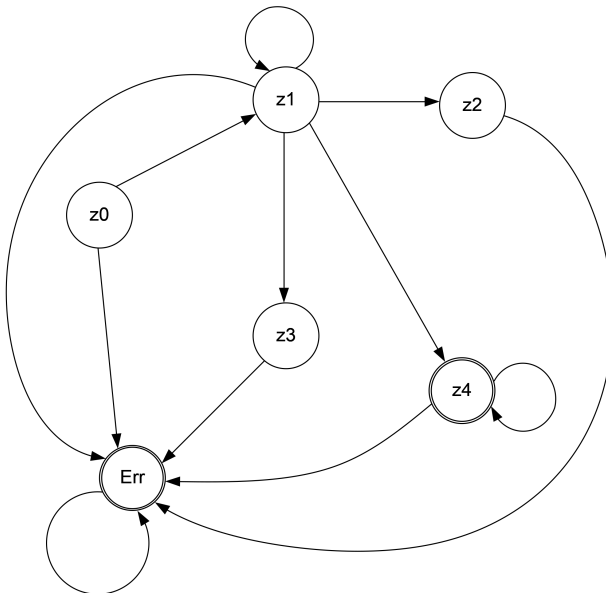


Таблица переходов автомата «Выражение»:

	z0	z1	z2	z3	z4	z5
б/ц	1	1	1	1	4	5
+	5	2	5	5	4	5
-	5	3	5	5	4	5
*,/,.	5	1	5	5	4	5
;	5	4	5	5	4	5
запр.	5	5	5	5	5	5

Граф переходов автомата «Выражение»:



Действия по состояниям:

z0: сброс

z1: передаем поступивший на вход символ автомату "Терм"

z2: терм считан, на вход поступил символ '+'

z3: терм считан, на вход поступил символ '-'

z4: финальное состояние

z5: состояние ошибки

Автомат «Терм»

Семантическая диаграмма:

ТЕРМ(a)::

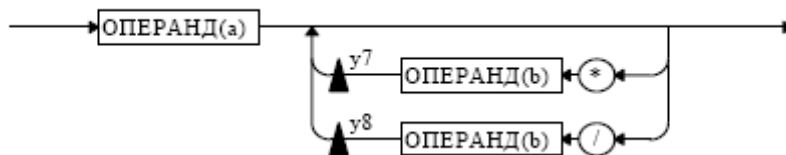
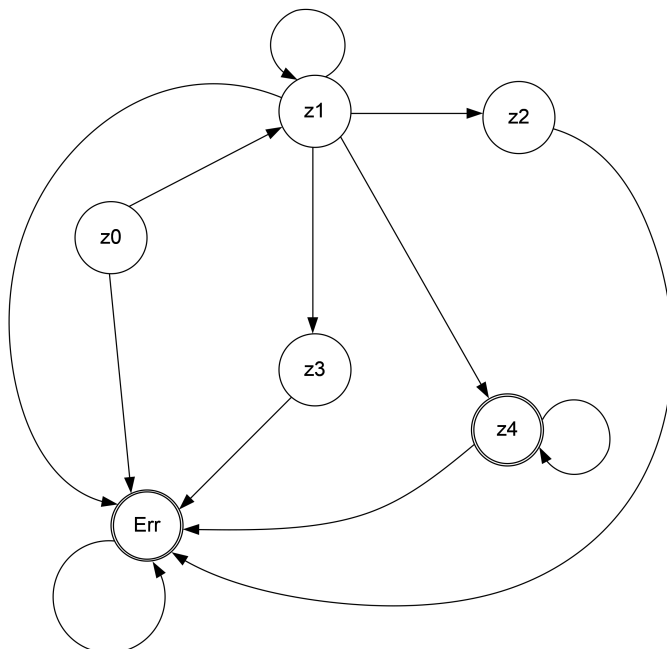


Таблица переходов автомата «Терм»:

	z0	z1	z2	z3	z4	z5
б/ц	1	1	1	1	4	5
*	5	2	5	5	4	5
/	5	3	5	5	4	5
.	5	1	5	5	4	5
;	5	4	5	5	4	5
запр.	5	5	5	5	5	5

Граф переходов автомата «Терм»



Действия по состояниям:

z0: сброс

z1: передаем поступивший на вход символ автомату "Операнд"

z2: операнд считан, на вход поступил символ '*'

z3: операнд считан, на вход поступил символ '/'

z4: финальное состояние

z5: состояние ошибки

Автомат «Операнд»

Семантическая диаграмма:

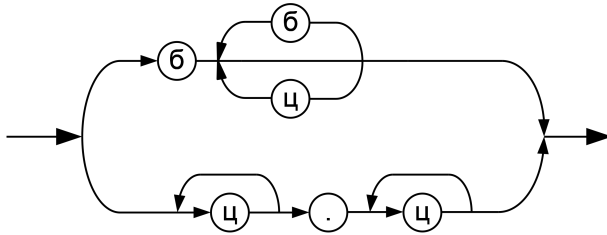
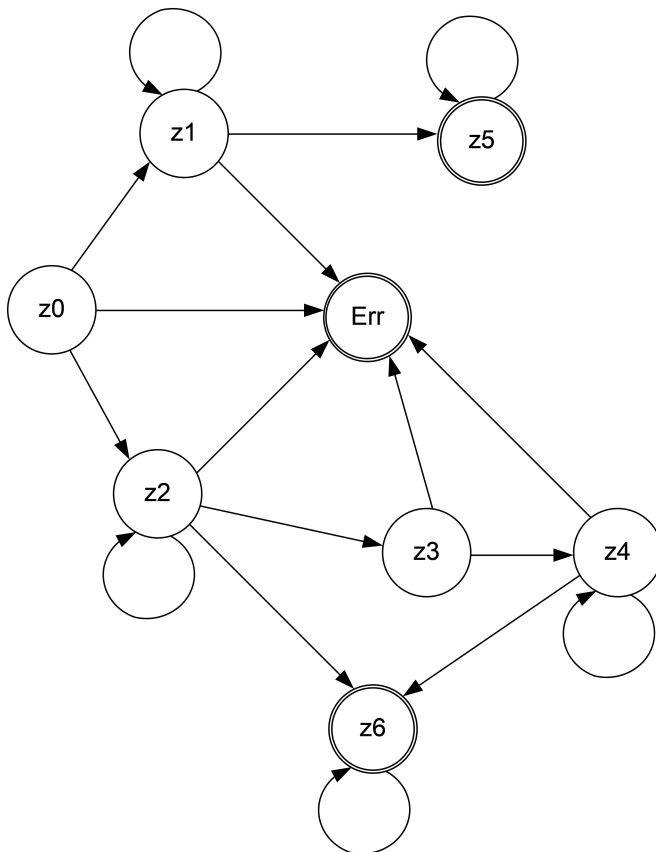


Таблица переходов автомата «Операнд»:

	z0	z1	z2	z3	z4	z5	z6	z7
буква	1	1	7	7	7	5	6	7
цифра	2	1	2	4	4	5	6	7
.	7	7	3	7	7	5	6	7
запр.	7	5	6	7	6	5	6	7

Граф переходов автомата «Операнд»:



Действия по состояниям:

z0: сброс

z1: добавляем поступивший на вход символ в имя переменной

z2: обрабатываем целую часть

z3: готовимся считывать дробную часть

z4: считываем дробную часть

z5: финальное состояние – переменная считана

z6: финальное состояние – число считано

z7: состояние ошибки