

SSQL 语言编译器测试报告

拟制 吕顺 2015 年 1 月 1 日

审阅 第 6 小组全体 2015 年 1 月 3 日

版本号	更新时间	更新内容
V1.0	2015 年 1 月 1 日	基本的语句识别与故障处理
V1.1	2015 年 1 月 2 日	组合主键和词法错误处理
V2.0	2015 年 1 月 3 日	逻辑的增加和词法的细节处理

目 录

一、	概 述	3
二、	产品需求	3
三、	测试准备	4
四、	测试内容迭代	5
五、	测试小结	17
六、	测试结果	17
七、	结果分析	18
八、	评估和建议	18

注：人员范围

本报告适用于 SSQL 编译器的开发人员、测试人员与项目经理阅读。

SSQL 语言 编译器测试报告

一、 概述

1. 系统概述

本软件实现的是一个 SSQL 语言的编译器，采用 C++ 语言编写，适当调用 STL 库，利用自动机原理和递归下降设计词法分析和语法分析器，得到一个建议的 SSQL 语言规则，并可以进行增删查的功能。

2. 文档概述

本文档包含测试的具体工作和迭代流程，以及应用最终版本代码产生的相应结果与预期相符程度，同时进行了适当的错误分析与改进建议。

二、 产品需求和 workflows

我们实现了课程设计要求的**全部基本项和加分项**，具体项目如下：

1.Context Free Grammar 实现基本的 CREATE、INSERT、SELECT、DELECT 操作

2.可以通过控制台输入，也可以通过文本文件输入

3.对于成功和失败的操作进行相应的提示（具体提示以 MySQL 为模板）

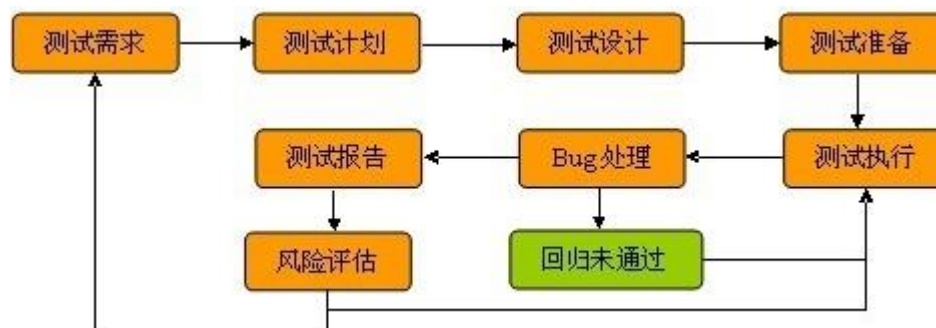
4.支持 expression 表达式，要求的算数、关系和逻辑操作符完美适配 (Bonus 1)

5.DEFAULT 语句和 VALUE LIST 的重载 (Bonus 2)

6. WHERE 从句和 BOOL 变量的重载 (Bonus 3)

7.创新点：在使用文件输入和控制台输入的时候会根据输入的语句错误给出相应提示（找不到文件路径、多余的错误字符等），还可以运行 help 查看帮助

在开发和测试的过程中，我们严格按照如下的流程进行开发，流水作业，并及时进行回归测试和迭代，得到了良好的效果：



三、 测试准备

a 测试时间：2015 年 1 月 1 日至 2015 年 1 月 3 日。

b 地点：中山大学东校区生活区。

c 硬件配置：CORE i5 内存 4.0G

d 软件配置：Wondows 8.1，编译器 VS2013（注：已实现对 g++编译的兼容）

e 被测软件版本号：V1.0，V1.1，V2.0

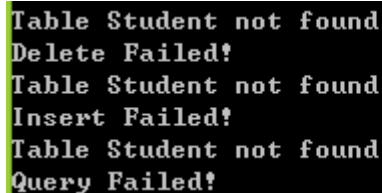
f 所有测试相关活动的日期和时间、测试操作人员等记录见软件测试记录文档。

四、 测试内容迭代

1.0基本测试（正常运行）

--不存在表时增删查改

```
DELETE FROM Student WHERE sid == 12330012;  
INSERT INTO Student(sid,sex,age) VALUES(12330001,0,10);  
SELECT sw, age FROM Student WHERE age <= 18;
```



```
Table Student not found  
Delete Failed!  
Table Student not found  
Insert Failed!  
Table Student not found  
Query Failed!
```

测试结果：符合预期！

--注：所有的语句，成功应当返回成功信息或表结果，失败应当返回失败信息

--一、CREATE 测试

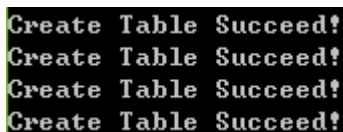
--1.建表

```
CREATE TABLE Student(sid INT,sex INT DEFAULT = 0, age INT DEFAULT = 18,  
PRIMARY KEY (sid));
```

```
CREATE TABLE _StudentM(sid INT,sex INT DEFAULT = 0, age INT DEFAULT = 18,  
PRIMARY KEY (sid));
```

```
CREATE TABLE Stu_de_ntN(sid INT,sex INT DEFAULT = 0, age INT DEFAULT = 18,  
PRIMARY KEY (sid));
```

```
CREATE TABLE StudentRR_(sid INT,sex INT DEFAULT = 0, age INT DEFAULT = 18,  
PRIMARY KEY (sid));
```

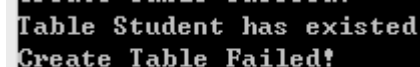


```
Create Table Succeed!  
Create Table Succeed!  
Create Table Succeed!  
Create Table Succeed!
```

测试结果：符合预期！

--2.建立已存在的表

```
CREATE TABLE Student(age INT DEFAULT = 18, sid INT,sex INT DEFAULT = 0,  
PRIMARY KEY (sid));
```



```
Table Student has existed  
Create Table Failed!
```

测试结果：符合预期！

--2.建立重复键值

```
CREATE TABLE Student(sex INT DEFAULT = 18, sid INT,sex INT DEFAULT = 0,  
PRIMARY KEY (sid));
```

```
near Line 10 column 88 : duplicate primary key
near Line 11 column 74 : duplicate primary key
```

测试结果：符合预期！

--4.建表中默认值的重载

```
CREATE TABLE Student3(sid INT,sex INT DEFAULT = 6 * 3 / 2 + 1, age INT DEFAULT =
2 * 3 - 6 / 2, PRIMARY KEY (sid));
```

```
CREATE TABLE Student4(sid INT,sex INT DEFAULT = -1, age INT DEFAULT = +12,
PRIMARY KEY (sid));
```

```
Create Table Succeed!
Create Table Succeed!
```

测试结果：符合预期！

二、INSERT 测试

--1.成功插入

```
INSERT INTO Student(sid,sex,age) VALUES(12330001,0,10);
```

```
INSERT INTO Student(sid,sex,age) VALUES(12330002,1,11);
```

```
INSERT INTO Student(sid,sex,age) VALUES(12330003,0,12);
```

```
Insert Succeed!
Insert Succeed!
Insert Succeed!
```

测试结果：符合预期！

--2.插入时各个字段名顺序调换

```
INSERT INTO Student(sex,age,sid) VALUES(1,13,12330004);
```

```
INSERT INTO Student(sex,sid,age) VALUES(0,12330005,14);
```

```
INSERT INTO Student(age,sid,sex) VALUES(15,12330006,1);
```

```
Insert Succeed!
Insert Succeed!
Insert Succeed!
```

测试结果：符合预期！

--3.插入已有的记录

```
INSERT INTO Student(sid,sex,age) VALUES(12330004,1,13);
```

```
Primary key value has exist
Insert Failed!
```

测试结果：符合预期！

--4.插入默认值记录

```
INSERT INTO Student(sid) VALUES(12330007);
```

```
INSERT INTO Student(sex,sid) VALUES(1,12330008);
```

```
INSERT INTO Student(sid,age) VALUES(12330009,(5*6)/2+4);
```

SSQL 语言 编译器测试报告

```
Insert Succeed!  
Insert Succeed!  
Insert Succeed!
```

测试结果：符合预期！

--5.插入语句 VALUE 值的重载

```
INSERT INTO Student(sid,sex,age) VALUES(12330010,1,6 * 3 / 2 + 1);
```

```
INSERT INTO Student(age,sid) VALUES(3*9-6/2,12330011);
```

```
INSERT INTO Student(sid,sex,age) VALUES(12330012,1,-333);
```

```
Insert Succeed!  
Insert Succeed!  
Insert Succeed!
```

测试结果：符合预期！

一三、SELECT 测试

--1.查询语句返回成功结果（成功）

```
SELECT sid, sex ,age FROM Student WHERE age < 18;
```

sid	sex	age
12330001	0	10
12330002	1	11
12330003	0	12
12330004	1	13
12330005	0	14
12330006	1	15
12330010	1	10
12330012	1	-333
12330013	0	3

9 rows affect.
Query Succeed!

测试结果：符合预期！

```
SELECT sid, sex ,age FROM Student WHERE age >= 19 && sex == 0;
```

SSQL 语言 编译器测试报告

sid	sex	age
12330009	0	19
12330011	0	24

2 rows affect.
Query Succeed!

测试结果：符合预期！

SELECT sid, sex ,age FROM Student WHERE age >= 19 || sex == 0;

sid	sex	age
12330001	0	10
12330003	0	12
12330005	0	14
12330007	0	18
12330009	0	19
12330011	0	24
12330013	0	3

7 rows affect.
Query Succeed!

测试结果：符合预期！

SELECT sid, sex ,age FROM Student WHERE age >= 19 && sex == 0 && sid == 12330009;

sid	sex	age
12330009	0	19

1 rows affect.
Query Succeed!

测试结果：符合预期！

SELECT sid, sex ,age FROM Student WHERE age >= 19 || sex == 0 && sid < 12330006;

SSQL 语言 编译器测试报告

sid	sex	age
12330001	0	10
12330003	0	12
12330005	0	14
12330009	0	19
12330011	0	24
5 rows affect. Query Succeed!		

测试结果：符合预期！

SELECT sid, sex ,age FROM Student WHERE age >= 19 || (sex == 0 && age < 13);

sid	sex	age
12330001	0	10
12330003	0	12
12330009	0	19
12330011	0	24
12330013	0	3
5 rows affect. Query Succeed!		

测试结果：符合预期！

SELECT sid ,age FROM Student;

SSQL 语言 编译器测试报告

sid	age
12330001	10
12330002	11
12330003	12
12330004	13
12330005	14
12330006	15
12330007	18
12330008	18
12330009	19
12330010	10
12330011	24
12330012	-333
12330013	3

13 rows affected.
Query Succeed!

测试结果：符合预期！

SELECT * FROM Student;

age	sex	sid
10	0	12330001
11	1	12330002
12	0	12330003
13	1	12330004
14	0	12330005
15	1	12330006
18	0	12330007
18	1	12330008
19	0	12330009
10	1	12330010
24	0	12330011
-333	1	12330012
3	0	12330013

13 rows affected.
Query Succeed!

测试结果：符合预期！

--2.查询语句中 WHERE 的重载

SELECT * FROM Student WHERE age + 7 > 19 + 6 && sex <> 6 / 3 - 1;

age	sex	sid
19	0	12330009
24	0	12330011

2 rows affect.
Query Succeed!

测试结果：符合预期！

--3.查询不存在的属性

SELECT sw, age FROM Student WHERE age <= 18;

SELECT sid,sex FROM Student WHERE ss == 1;

SELECT sw, FROM Student WHERE age >= 19 && sex == 0;

SELECT sid, FROM Student;

SSQL 语言 编译器测试报告

```
The select column sw not in the table
Query Failed!
The select column ssnot in the table
Query Failed!
near Line 86 column 16 : wrong type error
near Line 87 column 16 : wrong type error
```

测试结果：符合预期！

--4.查询不存在的记录

SELECT sid, age FROM Student WHERE age < -333;

```
-----|-----|
|      sid      |      age      |
|-----|-----|
0 rows affect.
Query Succeed!
```

测试结果：符合预期！

--5.查询不存在的表

SELECT sid, age FROM StudentSS WHERE age < 18;

SELECT * FROM Stu;

```
Table StudentSS not found
Query Failed!
Table Stu not found
Query Failed!
```

测试结果：符合预期！

一四、DELETE 测试

--1.各种删除语句（成功）

DELETE FROM Student WHERE age <= 18 && age > 14 && sex == 1;

SELECT * FROM Student;

SSQL 语言 编译器测试报告

2 rows affected.
Delete Succeed!

age	sex	sid
10	0	12330001
11	1	12330002
12	0	12330003
13	1	12330004
14	0	12330005
18	0	12330007
19	0	12330009
10	1	12330010
24	0	12330011
-333	1	12330012
3	0	12330013

11 rows affected.
Query Succeed!

测试结果：符合预期！

```
DELETE FROM Student WHERE sid == 12330012;
SELECT * FROM Student;
```

SSQL 语言 编译器测试报告

1 rows affected.
Delete Succeed!

age	sex	sid
10	0	12330001
11	1	12330002
12	0	12330003
13	1	12330004
14	0	12330005
18	0	12330007
19	0	12330009
10	1	12330010
24	0	12330011
3	0	12330013

10 rows affected.
Query Succeed!

测试结果：符合预期！

DELETE FROM Student WHERE sex == 1;
SELECT * FROM Student;

3 rows affected.
Delete Succeed!

age	sex	sid
10	0	12330001
12	0	12330003
14	0	12330005
18	0	12330007
19	0	12330009
24	0	12330011
3	0	12330013

7 rows affected.
Query Succeed!

测试结果：符合预期！

SSQL 语言 编译器测试报告

--2.删除语句中 WHERE 的重载

```
DELETE FROM Student WHERE age > 4*4 - 4 && sid == 6 / 3 - 2;
```

```
SELECT * FROM Student;
```

```
4 rows affected.
Delete Succeed!
+-----+-----+-----+
| age | sex | sid |
+-----+-----+-----+
| 10 | 0 | 12330001 |
+-----+-----+-----+
| 12 | 0 | 12330003 |
+-----+-----+-----+
| 3 | 0 | 12330013 |
+-----+-----+-----+
3 rows affected.
Query Succeed!
```

测试结果：符合预期！

```
DELETE FROM Student WHERE age + 10 > 24 - 4 && sex <> 1;
```

```
SELECT * FROM Student;
```

```
1 rows affected.
Delete Succeed!
+-----+-----+-----+
| age | sex | sid |
+-----+-----+-----+
| 10 | 0 | 12330001 |
+-----+-----+-----+
| 3 | 0 | 12330013 |
+-----+-----+-----+
2 rows affected.
Query Succeed!
```

测试结果：符合预期！

--3.删除不存在的记录

```
DELETE FROM StudentP WHERE age < 0;
```

```
DELETE FROM Student WHERE sid == 12330012;
```

```
DELETE FROM Student WHERE age < 18 && age > 14 && sex == 1;
```

```
Table StudentP not found
Delete Failed!
0 rows affected.
Delete Succeed!
0 rows affected.
Delete Succeed!
```

测试结果：符合预期！

--4.删除语义错误

```
DELETE FROM Student8 WHERE age + 7 > 14 + 6 && sex <> 6 / 3 - 2;
```

```
DELETE FROM Student WHERE nothing > 1 && sex <> 6 / 3 - 2;
```

```
Table Student8 not found
Delete Failed!
The select column nothingnot in the table
Delete Failed!
```

测试结果：符合预期！

--6.删除整张表

```
DELETE FROM Student;
SELECT * FROM Student;
```

```
Delete Succeed!
+-----+-----+-----+
|      age      |      sex      |      sid      |
+-----+-----+-----+
0 rows affected.
Query Succeed!
```

测试结果：符合预期！

1.1 附加测试（按语句测试）

编号	1.0 版行号	1.1 版行号	更改说明
1	19	22	增加建表时的操作测试
2	26	29	组合主键插入成功失败情况测试
4	95	98	注释变更
5	108 行后	113~116	增加新变量
6	171、172	180、181	命令字大小写变更

--1.建表时用到双甚至多主键

```
CREATE TABLE Employee123(group INT, num INT, age INT DEFAULT = 18, PRIMARY
KEY (group,num));
CREATE TABLE Employee(group INT, num INT, age INT DEFAULT = 18, sex INT
DEFAULT = 0, PRIMARY KEY (group,num,sex));
```

```
Create Table Succeed!
Create Table Succeed!
```

测试结果：符合预期！

--2.多主键时的插入

```
INSERT INTO Employee(group,num,sex,age) VALUES(01,1,0,10);
INSERT INTO Employee(group,num,sex,age) VALUES(01,1,1,10);
INSERT INTO Employee(group,num,sex,age) VALUES(01,2,1,10);
INSERT INTO Employee(group,num,sex,age) VALUES(01,3,2,10);
INSERT INTO Employee(group,num,sex,age) VALUES(01,4,0,10);
INSERT INTO Employee(group,num,sex,age) VALUES(02,1,0,10);
INSERT INTO Employee(group,num,sex,age) VALUES(03,1,0,10);
```

```
Insert Succeed!  
Insert Succeed!  
Insert Succeed!  
Insert Succeed!  
Insert Succeed!  
Insert Succeed!  
Insert Succeed!  
Insert Succeed!
```

测试结果：符合预期！

--3.组合主键插入失败

```
INSERT INTO Employee(group,num,sex,age) VALUES(01,1,0,10);  
INSERT INTO Employee(group,num,sex,age) VALUES(01,1,1,10);
```

```
Primary key value has exist  
Insert Failed!  
Primary key value has exist  
Insert Failed!
```

测试结果：符合预期！

2.0极端测试（词法、语法错误和边界）

编号	1.1 版行号	2.0 版行号	更改说明
1	19	22	注释变更
2	26	29	添加测试用例
3	29	32	添加测试用例

--3.建表语法错误

```
CREATE TABLE Stude77nt2(sid INT,sex, age INT DEFAULT = 18, PRIMARY KEY (sid));  
CREATE TABLE Stud5ent2(sid INT,abc DEFAULT = .18, age INT DEFA$ULT = */18,  
PRIMARY KEY (sid));
```

```
CREATE TABLE Stude6nt2(sid INT,s**( )x DEFAULT = 1, age INT DEFAULT = /18);  
CREATE TABLE StudentM sid INT,sex INT DEFAULT = 0, age INT DEFAULT = 18 ;  
CREATE TABLE StudentSDF(sid INT sex INT DEFAULT = 0, age INT DEFAULT = 18);
```

```
near Line 15 column 36 : syntax error  
near Line 16 column 42 : syntax error  
near Line 17 column 33 : syntax error  
near Line 18 column 25 : syntax error  
near Line 19 column 35 : syntax error
```

测试结果：符合预期！

--5.建表默认值遇到浮点数

```
CREATE TABLE Student5(sid INT,sex INT DEFAULT = 6 * 3 / 5, age INT DEFAULT = 32  
* 3 - 6 / 2, PRIMARY KEY (sid));
```



```
near Line 15 column 36 : syntax error
near Line 16 column 42 : syntax error
near Line 17 column 33 : syntax error
near Line 18 column 25 : syntax error
near Line 19 column 35 : syntax error
```

测试结果：符合预期！

--6.插入语句语法错误（不匹配）

```
INSERT INTO Student(sid) VALUES(12330013,1,+2);
INSERT INTO Student(sid) VALUES(12330013,7);
```

```
Insert Failed!
# of columns no equal to # of values
Insert Failed!
# of columns no equal to # of values
Insert Failed!
```

测试结果：符合预期！

--7.插入中遇到浮点数

```
INSERT INTO Student(sid,sex,age) VALUES(12330013,0,6 * 3 / 5);
INSERT INTO Student(sid,sex,age) VALUES(12330014 ,0,6 / 5);
INSERT INTO Student(sid,sex,age) VALUES(12330015,0,6 * 7 / 5/2);
```

```
Insert Succeed!
Insert Succeed!
Insert Succeed!
Insert Succeed!
```

测试结果：符合预期！

--6.查询语法错误

```
SELECT * FROM Student WHERE (age + 7) > 19 + 6 && sex <> 6 / 3 - 1;
```

```
near Line 97 column 37 : Wrong type error
```

测试结果：符合预期！

--5.删除语句语法错误

```
DELETE FROM Student WHERE nothing;
DELETE FROM Student WHAABB;
DELETE FROM Student WHERE age > (0) && (sid) == (6 / 3 - 2);
```

```
near Line 122 column 34 : Wrong type error
near Line 123 column 26 : syntax error
near Line 124 column 33 : Wrong type error
```

测试结果：符合预期！

--6.边界测试（数目超过 100）

（由于边界测试具体语句太长，会随文件一并打包附上，详情参见 test2.txt）

SSQL 语言 编译器测试报告

```
CREATE TABLE Student();
CREATE TABLE Student(sid INT = 0);
CREATE TABLE Student(INT sid = 0);
CREATE TABLE Student(sid INT, age INT DEFAULT);
CREATE TABLE Student(sid INT, age INT DEFAULT = 10, age INT);
CREATE TABLE Student(sid INT, age INT DEFAULT = 10, PRIMARY KEY (height));
CREATE TABLE Student(sid INT, PRIMARY KEY (sid), age INT DEFAULT = 10, PRIMARY KEY (age));
CREATE TABLE Student(sid INT, age INT DEFAULT = 10, PRIMARY KEY (sid, age));
CREATE TABLE Student(sid INT);
CREATE TABLE S(a INT, a0 INT, a1 INT, a2 INT, a3 INT, a4 INT, a5 INT, a6 INT, a7 INT, a8 INT, a9
INT, a10 INT, a11 INT, a12 INT, a13 INT, a14 INT, a15 INT, a16 INT, a17 INT, a18 INT, a19 INT, a20
INT, a21 INT, a22 INT, a23 INT, a24 INT, a25 INT, a26 INT, a27 INT, a28 INT, a29 INT, a30 INT, a31
INT, a32 INT, a33 INT, a34 INT, a35 INT, a36 INT, a37 INT, a38 INT, a39 INT, a40 INT, a41 INT, a42
INT, a43 INT, a44 INT, a45 INT, a46 INT, a47 INT, a48 INT, a49 INT, a50 INT, a51 INT, a52 INT, a53
INT, a54 INT, a55 INT, a56 INT, a57 INT, a58 INT, a59 INT, a60 INT, a61 INT, a62 INT, a63 INT, a64
INT, a65 INT, a66 INT, a67 INT, a68 INT, a69 INT, a70 INT, a71 INT, a72 INT, a73 INT, a74 INT, a75
INT, a76 INT, a77 INT, a78 INT, a79 INT, a80 INT, a81 INT, a82 INT, a83 INT, a84 INT, a85 INT, a86
INT, a87 INT, a88 INT, a89 INT, a90 INT, a91 INT, a92 INT, a93 INT, a94 INT, a95 INT, a96 INT, a97
INT, a98 INT, a99 INT, a100 INT, a101 INT);
CREATE TABLE S(a INT, a0 INT, a1 INT, a2 INT, a3 INT, a4 INT, a5 INT, a6 INT, a7 INT, a8 INT, a9
INT, a10 INT, a11 INT, a12 INT, a13 INT, a14 INT, a15 INT, a16 INT, a17 INT, a18 INT, a19 INT, a20
INT, a21 INT, a22 INT, a23 INT, a24 INT, a25 INT, a26 INT, a27 INT, a28 INT, a29 INT, a30 INT, a31
INT, a32 INT, a33 INT, a34 INT, a35 INT, a36 INT, a37 INT, a38 INT, a39 INT, a40 INT, a41 INT, a42
INT, a43 INT, a44 INT, a45 INT, a46 INT, a47 INT, a48 INT, a49 INT, a50 INT, a51 INT, a52 INT, a53
INT, a54 INT, a55 INT, a56 INT, a57 INT, a58 INT, a59 INT, a60 INT, a61 INT, a62 INT, a63 INT, a64
INT, a65 INT, a66 INT, a67 INT, a68 INT, a69 INT, a70 INT, a71 INT, a72 INT, a73 INT, a74 INT, a75
INT, a76 INT, a77 INT, a78 INT, a79 INT, a80 INT, a81 INT, a82 INT, a83 INT, a84 INT, a85 INT, a86
INT, a87 INT, a88 INT, a89 INT, a90 INT, a91 INT, a92 INT, a93 INT, a94 INT, a95 INT, a96 INT, a97
INT, a98 INT, a99 INT, a100 INT, a101 INT, PRIMARY KEY (a, a0, a1, a2, a3, a4, a5, a6, a7, a8, a9,
a10, a11, a12, a13, a14, a15, a16, a17, a18, a19, a20, a21, a22, a23, a24, a25, a26, a27, a28, a29,
a30, a31, a32, a33, a34, a35, a36, a37, a38, a39, a40, a41, a42, a43, a44, a45, a46, a47, a48, a49,
a50, a51, a52, a53, a54, a55, a56, a57, a58, a59, a60, a61, a62, a63, a64, a65, a66, a67, a68, a69,
```

```
# of columns in the table S > 100.
Create Table Failed!
# of columns in primary key declaration > 100 in the table S
Create Table Failed!
```

测试结果：符合预期！

五、 测试小结

本次测试对 SSQL 的编译器进行了静态分析和动态测试。测试工作分为两个阶段。第一阶段进行了软件静态分析，软件测试人员和开发人员分别对软件 V1.0 版本的代码进行走读。在此基础上软件开发人员对代码走查中发现问题进行了修改，做了 97 处代码变更并提交 V1.1 版本进行动态测试（具体代码变更处就不详细列出），改进后提交了 V2.0 版本。

在测试过程中针对发现的软件缺陷进行了初步分析，并提交程序设计人员对原软件中可能存在的问题进行考查。在软件测试中首先根据软件测试的规范进行考核，将书写规范，注释等基础问题首先解决，其次考核软件测试中的问题是否存在设计上的逻辑缺陷，如果存在设计缺陷则应分析该缺陷的严重程度以及可能引发的故障。软件开发人员在以上基础上对软件的不足做出相应的修改，同时通过软件回归测试验证软件修改后能够得到的改善结果。同时使用变测试边迭代的方法，加快了作业的流程，提高了项目的整体性

六、 测试结果

在两个阶段测试过程中共发现软件缺陷 36 个，经软件开发人员确认的缺陷为 25 个，经过改正的代码消除了所有以确认的软件缺陷并通过了回归测试。因测试条件所限，未能进行软件的确认测试和系统测试。

静态测试采用人工代码走查的方式进行。参加代码走查的软件开发人员有：于德强，杨靖，杨浦城，袁伟佳；参加代码走查的软件测试人员有：吕顺。代码走查以代码审查会议的形式进行。静态分析过程中共进行了四次会议审查。静态测试阶段的主要工作内容是：

SSQL 语言 编译器测试报告

根据对软件汇编源代码的分析绘制详细的程序流程图和调用关系图（见附件中的设计文档）；

对照软件汇编源代码和流程图进行程序逻辑分析、算法分析、结构分析和接口分析；
对软件汇编源代码进行编程规范化分析。

通过静态测试查找出软件的缺陷 18 个，其中：

轻微的缺陷 10 个，占有所有缺陷的 26.2%

中等的缺陷 18 个，占有所有缺陷的 50.0%

严重的缺陷：8 个，占有所有缺陷的 23.7%

总共的测试用例数：143 个。全部由测试人员人工设计。

发现的软件缺陷有 2 个，都是在单元测试过程中发现的。集成测试阶段未发现新的软件缺陷。在发现的软件缺陷中：

中等的缺陷 1 个，占有所有缺陷的 50%

严重的缺陷 1 个，占有所有缺陷的 50%

动态测试中代码覆盖率达到百分之百，最终版本的代码完全达到要求：

代码行覆盖率	100%
分支覆盖率	100%
程序单元调用覆盖率	100%

七、 测试分析

下面将对此次软件测试中的所有缺陷以及改进设计进行分析。

静态测试中的缺陷分析：

4 个轻微缺陷属于代码冗余，由于在程序设计中加入了部分调试程序，在程序设计完成后未将这些调试代码注释或删除掉而造成代码冗余，但对程序本身的功能并无影响。修改后程序的效率得到提高。

18 个中等缺陷属于注释变更，在原程序代码的注释中存在注释不准确的问题，会影响程序员对程序的理解，修改后的程序提高了程序的可读性。

重点分析 3 个严重缺陷：

其中有由于代码的**逻辑不严密出现的崩溃问题**，也有由于开发人员沟通不畅导致的**传入参数问题**和结果预期不符，在单元测试的部分完全符合预期，但在最后动态测试阶段出现了问题，还有一些是由于**编译器的不兼容性**导致的符号从错误。

动态测试中的缺陷分析：

中等缺陷 1 个，在程序的逻辑判断中应当添加条件，修改后问题得到了解决，提高了程序的可读性。

严重缺陷 2 个，在无效的判别中，本应判断大于 10，但误设计为 0，修改后经回归测试问题得到了解决。 还有临时添加的用命令行输入的问题导致的出现后续问题。

改进的设计分析：由于涉及到产品设计问题，在此不加赘述。

八、 评估和建议

1. 测试评估

经过回归测试，未残留的软件编码规范性缺陷。软件代码文本注释率约为 20%，代码注释还欠充分，代码的理解和维护性上有待加强。

被测软件单元的总数：80 个

SSQL 语言 编译器测试报告

使用的测试用例个数：143 个

达到软件测试出口准则的软件单元数为 80 个，通过率 10%

通过单元和集成测试得知：软件代码逻辑清晰、结构合理、程序单元间接口关系一致，运行稳定。运行稳定的系统

2. 改进建议

a. 建议在软件开发项目中全面实施软件工程化，加强软件开发的管理工作。

b. 建议进一步加强软件需求规格说明、软件设计文档编制以及编写代码的规范化。特别是应该将系统中的硬件研制和软件研制分别管理，软件文档编制的种类和规格按照相关标准执行。

c. 尽早开展软件测试工作。在软件研制计划安排上给软件测试留有必要的时间，在资源配置上给软件测试必要的支撑。

d. 建议结合系统联试，开展软件的确认和系统测试。

e. 最重要的是要细致地阅读和浏览文档，找出每一步具体细致的需求，才能够提高代码健壮性，以不变应万变。