

# C三三 - 测试报告

王晨宇 3180104919

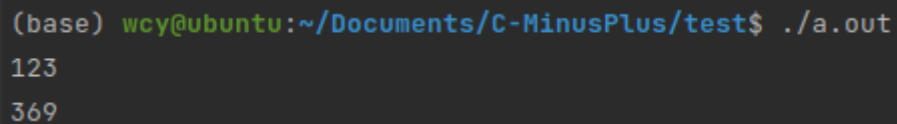
本测试报告就实验报告中提到的编译器基本功能和错误检测进行测试，所有测试代码都可以在文件夹中找到

## 一、基本功能测试

### 1.1 输入输出

1. test/io.c

```
int main() {  
    int a; int b; int c;  
    a = read();  
    print("%d", a * 3);  
}
```



```
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out  
123  
369
```

### 1.2 变量定义和作用域

test/scope.c

```
int a; int b;  
int func(int a, int b) {  
    print("func a = %d, b = %d", a, b);  
    if (a > b) {  
        int b; b = a;  
        print("func: b = %d", b);  
    }  
    else {  
        int a; a = b;  
        print("func: a = %d", a);  
    }  
    print("exit func : %d %d", a, b);  
}  
int main() {  
    a = read(); b = read();  
    print("main read: %d %d", a, b);  
    if (a < b) {  
        int b; b = a;  
        print("main: min b = %d", b);  
    }  
    else {  
        int a; a = b;  
        print("main : min a = %d", a);  
    }  
}
```

```

func(a, b);
print("exit main :%d %d", a, b);
}

```

```

(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
10
20
main read: 10 20
main: min b = 10
func a = 10, b = 20
func: a = 20
exit func : 10 20
exit main :10 20

```

### 1.3 四则运算

test/cal.c.c

```

int main() {
    int i;
    int j;
    int k;
    int result;
    i = 1; j = 11; k = 39;
    result = 0;
    i = (-i) + k + -(17 * i);
    print("%d", i);
    j = 11 * j - (k + i * i) * i;
    print("%d", j);
    k = i * (i / j) + k;
    print("%d", k);
    result = 4 * i + j / 17 + i * k;
    print("%d", result);
    return 0;
}

```

这里对于不同的编译语言定义和不同运行环境可能有不同的结果。例如南大测试手册给的预期结果是 21 -9959 18 -124。而我的代码和gcc的结果保持一致。

```

(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
21
-9959
39
318
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ gcc test1.c -o std
test1.c: In function 'main':
test1.c:9:5: warning: implicit declaration of function 'printf' [-Wimplicit-function-declaration]
    printf("%d", i);
    ^~~~~~
test1.c:9:5: warning: incompatible implicit declaration of built-in function 'printf'
test1.c:9:5: note: include '<stdio.h>' or provide a declaration of 'printf'
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./std
21-995939318(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$

```

## 1.4 if语句

test/branch.c

一个输入两个对角顶点的坐标判断两个矩形的公共面积是否大于 0 的小程序,

```
int main() {
    int x1; int y1; int x2; int y2;
    int u1; int v1; int u2; int v2;
    int l1; int r1; int t1; int b1;
    int l2; int r2; int t2; int b2;
    x1 = read();
    y1 = read();
    x2 = read();
    y2 = read();
    u1 = read();
    v1 = read();
    u2 = read();
    v2 = read();
    if (x1 == x2 || y1 == y2 || u1 == u2 || v1 == v2) {
        print("%d", -1);
    } else {
        if (x1 < x2) {
            l1 = x1;
            r1 = x2;
        } else {
            l1 = x2;
            r1 = x1;
        }
        if (y1 < y2) {
            t1 = y2;
            b1 = y1;
        } else {
            t1 = y1;
            b1 = y2;
        }
        if (u1 < u2) {
            l2 = u1;
            r2 = u2;
        } else {
            l2 = u2;
            r2 = u1;
        }
        if (v1 < v2) {
            t2 = v2;
            b2 = v1;
        } else {
            t2 = v1;
            b2 = v2;
        }

        if (l2 >= r1 || r2 <= l1 || b2 >= t1 || t2 <= b1) {
            print("%d", 0);
        } else {
            print("%d", 1);
        }
    }
}
```

```
    return 0;
}
```

由于输入较为复杂在此不展示输出结果。

测试结果均正确

## 1.5 while语句

test/loop.c

计算杨辉三角中第  $n$  个数所在的行，行中的位置，以及数值

```
int main() {
    int k;
    int line; int cnt;
    int m; int n;
    int numtor; int denomtor;
    numtor = 1;
    denomtor = 1;
    k = read();

    if (k <= 0) {
        print("%d", -1);
        return 0;
    }

    while (cnt < k) {
        line = line + 1;
        cnt = cnt + line;
    }
    m = k - (cnt - line);
    n = line;
    print("%d", n);
    print("%d", m);

    cnt = 0;
    while (cnt < m) {
        numtor = numtor * n;
        n = n - 1;
        cnt = cnt + 1;
    }
    while(m > 0) {
        denomtor = denomtor * m;
        m = m - 1;
    }
    print("%d", numtor / denomtor);
    return 0;
}
```

```
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
5
3
2
3
```

```
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
17
6
2
15
```

## 1.6 函数调用

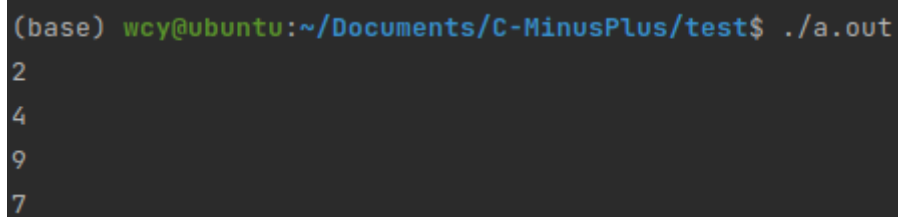
### 1. test/fpow.c

快速幂：计算 $a^b \bmod c$ 的值

```
int mod(int i, int j) {
    return i - i / j * j;
}

int quick_power_mod(int x, int y, int k) {
    int res; res = 1;
    if (x <= 0 || y <= 0 || k <= 0) {
        return -1;
    } else {
        x = mod(x, k);
        while (y != 0) {
            if (mod(y, 2) == 1) {
                res = mod(res * x, k);
            }
            y = y / 2;
            x = mod(x * x, k);
        }
        return res;
    }
}

int main() {
    int a; int b; int c; int cnt;
    a = read();
    b = read();
    c = read();
    printf("%d", quick_power_mod(a, b, c));
    return 0;
}
```



```
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
2
4
9
7
```

$2^4 \equiv 7 \pmod{9}$

### 2. test/fac\_prime.c

```
int fact(int m) {
    if (m <= 0) {
        return 1;
    } else {
        return fact(m - 1) * m;
    }
}
```

```

int isqrt(int n) {
    int i;
    i = 0;
    while (i < n) {
        if (i * i <= n && (i + 1) * (i + 1) > n) {
            return i;
        }
        i = i + 1;
    }
    return -1;
}

int mod(int k1, int k2) {
    if (k1 < 0 || k2 <= 0) {
        return -1;
    } else {
        return k1 - k1 / k2 * k2;
    }
}

int is_prime(int l) {
    int j; int end;
    j = 2;
    end = isqrt(l);
    while (j <= end) {
        if (mod(l, j) == 0) {
            return 0;
        }
        j = j + 1;
    }
    return 1;
}

int main() {
    int c; int d;
    c = 2;
    d = read();
    while (c < fact(d)) {
        if (is_prime(c)) {
            print("%d", c);
        }
        c = c + 1;
    }
    return 0;
}

```

```
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
0
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
3
2
3
5
(base) wcy@ubuntu:~/Documents/C-MinusPlus/test$ ./a.out
4
2
3
5
7
11
13
17
19
23
```

## 二、错误测试

### 2.1 词法分析

#### 1. 未识别的字符

error/wrongtoken.c

```
int main() {
    int a;
    a = #;
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c

Syntax Error at line:3 Col:9,
    Unrecognized character: #

Syntax Error at line:3 Col:10,
    syntax error
Syntax error

Process finished with exit code 0
```

#### 2. 不规范的注释

error/comment.c

```
// this is a comment
int main() {
    /**
     * This is a multi row comment
     */ // 文件中没有第一个星号
    int a;
    a = 10;
    print("%d", a);
}
```

因为错误的注释会搞乱markdown的语法高亮所以在这里还是把注释写对了

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c

Syntax Error at line:9 Col:6,
    Syntax Error: Improper Comments

Syntax Error at line:9 Col:6,
    syntax error
Syntax error

Process finished with exit code 0
```

## 2.2 语法分析

### 2.2.1 变量定义

1. 变量名不能与函数名相同

error/samevar.c

```
int a() {
}
int main() {
    int a;
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c
Syntax error: line: 5, col: 10;
    Function a() has been declared!
```

2. 不允许同名变量在同一作用域中重复定义

error/samevar.c



```
int main() {
    int a;
    int a;
    fun();
}
int fun() {
    int b; int b;
}
```

```
Syntax error: line: 3, col: 10;
    Variable a has been declared!
Syntax error: line: 4, col: 10;
    function fun has not been declared.
Syntax error: line: 7, col: 17;
    Variable b has been declared!
```

### 3. 不允许同名函数重复定义

error/samefunc.c

```
int main() {
}
int a() {

}
int a() {

}
```

```
Syntax error: line: 6, col: 7;
    function 'a' has been declared before.

Process finished with exit code 0
```

### 4. 变量使用前必须定义

error/undefinedvar.c

```
int main() {
    int a;
    b = 10;
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c
Syntax error: line: 3, col: 7;
    Variable b has not been declared!
```

### 5. 函数调用前必须定义

error/undefindfunc.c

```
int func() {
    return "123";
}
int main() {
    a();
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c
Syntax error: line: 2, col: 17;
    function 'func' returns type int but here returns string
Syntax error: line: 5, col: 8;
    function a has not been declared.
```

如调用另一个函数，则被调用函数必须在调用函数之前定义，如：

error/fuck.c

```
int f() {
    return g();
}
int g() {
    return f();
}
int main() {
    f();
    g();
}
```

```
Syntax error: line: 2, col: 14;
    function g has not been declared.
Syntax error: line: 2, col: 15;
    error occur in 'g()'
```

6. 没有定义main () 方法; 不能调用main()方法

error/main.c

```
int main() {
    f();
}

int e() {
    return main();
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c
Syntax error: line: 3, col: 8;
    function f has not been declared.
Syntax error: line: 7, col: 17;
    can't call function 'main'
```

error/nomain.c

```
int a() {  
  
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c  
Syntax error: line: 3, col: 1;  
undefined entrance 'main'
```

### 2.2.2 类型检查

#### 1. 赋值运算两边类型必须相同

error/diff\_type.c

```
int main() {  
    int a;  
    a = "123";  
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c  
Syntax error: line: 3, col: 14;  
Can't assign string to int.
```

虽然变量类型不支持字符串，但是字符串还是可以作为常量被程序识别。

#### 2. 赋值运算左边必须是左值

error/left.c

```
int func() {  
  
}  
int main() {  
    int a;  
    a + 4 = 123;  
    func() = 123;  
    func = 123;  
}
```

```
Syntax error: line: 6, col: 16;  
    a+4 is not a left value!  
Syntax error: line: 7, col: 17;  
    func() is not a left value!  
Syntax error: line: 8, col: 10;  
    Variable func has not been declared!  
  
Process finished with exit code 0
```

#### 2. 函数调用参数数量和类型必须顺序相同

test\funcall.c

```
int func() {
    return "123";
}
int main() {
    func(1);
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c
Syntax error: line: 2, col: 17;
    function 'func' returns type int but here returns string
Syntax error: line: 5, col: 12;
    function func requires 0 arguments, but 1 provided
```

### 3. 函数返回类型与定义类型必须相同

test\ret\_type.c

```
int main() {
}
int func() {
    return "123";
}
```

```
/home/wcy/Documents/C-MinusPlus/C_MinusPlus test/test1.c
Syntax error: line: 5, col: 17;
    function 'func' returns type int but here returns string
```