

- 作业1-2语言定义分析报告
 - 逐文件分析
 - test1
 - test2
 - test3
 - test4
 - test5
 - test6
 - 总语法规则

作业1-2语言定义分析报告

逐文件分析

test1

src:

```
int get(int a[], int b) {
    int ret = 0;
    int i = 0;
    while (i < 6) {
        ret = ret + a[i] / b + a[i] % b;
        i = i + 1;
    }
    return ret;
}

int id = 21052021;

int main() {
    putint(id);
    putch(10);

    int a[6] = {5, 0, 0, 40, 48, 28};
    int b[6] = {1, 0, 0, 2, 0, 1};

    int ptr = 0;
    int all = 0;
    while (ptr < 6) {
        all = all + a[ptr] - 20 * b[ptr];
        ptr = ptr + 1;
    }
}
```

```

    // printf("%d\n", all);
    putint(all);
    putchar(10);

    int t = get(a, 5);

    putint(t);
    putchar(10);
    return 0;
}

```

涵盖语法规则:

- 语句块
- 语句while\return\赋值\函数调用
- 整数数值
- 表达式 +,-,*,/,%,<
- 形参 int a[], int b
- 变量初值

test2

src:

```

int check(int val) {
    if (val >= 100) return 5;
    else if (val >= 80) return 4;
    else if (val >= 60) return 3;
    else if (val > 0) return 2;
    else if (val == 0) return 1;
    else return 0;
}

const int id = 21052021;

int main() {
    putint(id);
    putchar(10);

    int a[20] =
{5,0,0,40,48,28,60,70,80,65,67,78,88,79,100,101,102,-1,-2,-2};

    int cnt[6] = {};

    int i = 0;
    while (i < 20) {
        int val = check(a[i]);
    }
}

```

```

        cnt[val] = cnt[val] + 1;
        i = i + 1;
    }

    int sum = 20;

    float rato = 0.0;
    rato = cnt[5] + cnt[4] + cnt[3];
    rato = rato / sum;

    putfloat(rato);
    putchar(10);

    return 0;
}

```

涵盖语法规则:

- 常量定义
- 语句块
- 语句while\return\赋值\函数调用
- 整数数值
- 加减乘除表达式 +,-,/
- 关系表达式>=,>,==
- 形参 int val
- 函数返回值 int
- 变量定义 float, int

test3

src:

```

int mul(int a, int b, int m) {
    int res = 0;
    while (b) {
        if (b % 2) res = (res + a) % m;
        b = b / 2;
        a = (a + a) % m;
    }
    return res;
}

int power(int a, int b, int m) {
    int res = 1 % m;
    while (b) {
        if (b % 2) res = mul(res, a, m);
    }
}

```

```

        b = b / 2;
        a = mul(a, a, m);
    }
    return res;
}

int ctz(int n);

int power1(int a, int b) {
    int res = 1;
    while (b) {
        if (b % 2) res = res * a;
        b = b / 2;
        a = a * a;
    }
    return res;
}

int isprime(int n) {
    if (n < 2) return 0;
    int A[9] = {2, 3, 5, 7, 11, 13, 17, 19, 23};
    int s = ctz(n - 1);
    int m = power1(2, s);
    int d = (n - 1) / m;
    int i = 0;
    while (i < 9) {
        int a = A[i];
        if (a == n) return 1;
        int x = power(a, d, n);
        if (x == 1 || x == n - 1) {
            i = i + 1;
            continue;
        }
        int ok = 0;

        int j = 0;
        while (s - 1 > j) {
            x = mul(x, x, n);
            if (x == n - 1) {
                ok = 1;
                break;
            }
            j = j + 1;
        }

        if (!ok) return 0;
        i = i + 1;
    }
    return 1;
}

```

```

// 米勒判素
int main() {
    int id = 21052021;
    putint(id);
    putch(10);
}

```

```

    int x;
    x = getint();

    int res = isprime(x);

    if (res) {
        putint(1);
        putch(10);
    } else {
        putint(0);
        putch(10);
    }
    return 0;
}

int ctz(int n) {
    int res = 0;
    while (n && n % 2 == 0) {
        res = res + 1;
        n = n / 2;
    }
    return res;
}

```

涵盖语法规则:

- 函数声明 `int ctz(int n);`
- 语句块
- 语句 `while`, `return`, `continue`, `break`, `if`, `else`, 赋值, 函数调用
- 整数数值
- 加减乘除表达式 `+`, `-`, `/`
- 关系表达式 `<`, `==`
- 逻辑与或表达式 `&&`, `||`
- 一元表达式 `!ok`
- 形参 `int val`
- 函数返回值 `int`
- 变量定义 `float`, `int`

test4

src:

```

int fib(int n) {
    if (n <= 0) return 0;
}

```

```

    if (n == 1) return 1;
    return fib(n - 1) + fib(n - 2);
}

void solve() {
    int x;
    x = getint();

    int res = fib(x);

    putint(res);
    putchar(10);
}

int main() {
    int id = 21052021;
    putint(id);
    putchar(10);

    int t;
    t = getint();
    while (t) {
        solve();
        t = t - 1;
    }

    return 0;
}

```

涵盖语法规则:

- 函数声明 `int fib(int n);`
- 语句块
- 语句 `while`, `return`, `if`, 赋值, 函数调用(递归)
- 整数数值
- 加减乘除模表达式 `+`, `-`
- 关系表达式 `<=`, `==`

test5

src:

```

float abs(float x) {
    if (x < 0) return -x;
    else return x;
}

//findsqrt(x) using binary search

```

```

float binary_search(float x) {
    if (x < 0) {
        return -1;
    }

    float l = 0, r = x;
    int round = 0;
    while (abs(r - l) > 1e-6 || round < 20) {
        float mid = (l + r) / 2;
        if (mid * mid <= x) {
            l = mid;
        } else {
            r = mid;
        }
        round = round + 1;
    }
    return l;
}

int main() {
    int id = 21052021;
    putint(id);
    putchar(10);

    float x;
    x = getfloat();

    float sqrt = binary_search(x);
    printfloat(sqrt);
    putchar(10);

    return 0;
}

```

涵盖语法规则:

- 语句块
- 语句while,return,if,else,赋值,函数调用
- 整数数值
- 浮点数常量1e-6
- 加减乘除表达式 -, *
- 关系表达式<,<=,>
- 逻辑与或表达式||
- 一元表达式-x
- 形参 float x
- 函数返回值 float
- 变量定义 float, int

test6

src

```
int A[2][4] = {{1, 2, 3, 4}, {4, 3, 2, 1}};
int B[4][2] = {};

int main() {
    int id = 21052021;
    putint(id);
    putchar(10);

    int x = 0;
    while (x < 8) {
        B[x % 4][x / 4] = A[x / 4][x % 4];
    }
    return 0;
}
```

涵盖语法规则:

- 语句块
- 语句while,return,赋值
- 整数数值
- 加减乘除模表达式 %, /
- 左值表达式 A[x / 4][x % 4]

总语法规则

- 函数声明 int ctz(int n);
- 语句块
- 语句while,return,continue,break,if,else,赋值,函数调用(递归)
- 整数数值
- 加减乘除模表达式 +, -, /, *, %
- 关系表达式 <, <=, ==, >, >=
- 逻辑与或表达式 &&, ||
- 一元表达式 !ok, -x
- 形参 int val, float x, int a[]
- 函数返回值 int, float, void
- 变量定义 float, int

- 左值表达式 $A[x / 4][x \% 4]$
- 浮点数常量 $1e-6$
- 常量定义
- 表达式 $+, -, *, /, \%, <$
- 变量初值

应该是都涵盖了