భారతీయ సాంకేతిక విజ్ఞాన సంస్థ హైదరాబాద్
भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Compilers-II

CPlex

language-specification

By Group-12

INDEX

1 Introduction

1.1 Motivation

motivation here

1.2 Goal

goal here

2 Data types

Hello world

3 Operators and expressions

Hello world

4 Lexical specifications

Hello world

5 Declarations

Hello world

6 Statements

Hello world

7 Built-in functions

7.1 Inbuilt complex functions:

- `real double : cdouble c`: Returns the real part of the complex number `c`.
- `img double : (cdouble c)`: Returns the imaginary part of the complex number `c`.
- `pow cdouble : (cdouble base,double exponent)`: Returns the complex number $(base)^{(exponent)}$. This is done by using De Moivre's formula.
- `polar void :(cdouble c)`: Prints the polar form of a complex number `c`. Given a complex number $c = a + ib$ the polar form looks $c = r(e^{i\theta})$ (Where θ is the argument of the complex number and r is the modulus of the complex number).
- `conjugate cdouble : (cdouble c)`: Returns the conjugate of the complex number `c`. Given a complex number $c = a + ib$ the conjugate looks like $c = a - ib$.

- `mod double : (cdouble c)`: Returns the modulus of the complex number `c`. Given a complex number $c = a + ib$ the modulus looks like $c = \sqrt{a^2 + b^2}$.

- `arg double :(cdouble c)`: Returns the argument of the complex number `c`. Given a complex number $c = a + ib$ the argument looks like $c = \tan^{-1}(\frac{b}{a})$.

- `angle double : (cdouble c1,cdouble c2)`: Returns the angle between the complex numbers `c1` and `c2`. Given two complex numbers $c_1 = a_1 + b_1 i$ and $c_2 = a_2 + b_2 i$ the angle between them looks like $c = \tan^{-1}(\frac{b_2 - b_1}{a_2 - a_1})$.

- `dist double : (cdouble c1,cdouble c2)`: Returns the distance between the complex numbers `c1` and `c2`. Given two complex numbers $c_1 = a_1 + b_1 i$ and $c_2 = a_2 + b_2 i$ the distance between them looks like $c = \sqrt{(a_2 - a_1)^2 + (b_2 - b_1)^2}$.

- `cprint void: (cdouble c)`: Prints the complex number `c` in the form $a + ib$.

7.2 Geometry related:

- `rotate cdouble : (cdouble c,cdouble origin,double angle)`: Returns the complex number `c` rotated by an angle `angle` about the point `origin`. The rotation is done in the counter-clockwise direction.

- `dist double :(cdouble c1,cdouble c2)`: Returns the distance between the complex numbers `c1` and `c2`. Given two complex numbers $c_1 = a_1 + b_1 i$ and $c_2 = a_2 + b_2 i$ the distance between them looks like $c = \sqrt{(a_2 - a_1)^2 + (b_2 - b_1)^2}$.

- `get_line void :(cdouble c1,cdouble c2,double *a,double *b,double *c)`: Given two complex numbers $c_1 = a_1 + b_1 i$ and $c_2 = a_2 + b_2 i$ this function prints the line $ax + by + c = 0$ passing through the points c_1 and c_2.

- `is_traingle bin :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns true if the points c_1,c_2 and c_3 form a triangle else false.

- `get_centroid cdouble:(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the centroid of the triangle formed by(if exists) the points c_1,c_2 and c_3.

- `get_circumcenter cdouble :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the circumcenter of the triangle formed by(if exists) the points c_1,c_2 and c_3.

- `get_orthocenter cdouble :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the orthocenter of the triangle formed by(if exists) the points c_1,c_2 and c_3.

- `get_incenter cdouble :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the incenter of the triangle formed by(if exists) the points c_1,c_2 and c_3.

- `get_excenter cdouble :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the excenter of the triangle formed by(if exists) the points c_1,c_2 and c_3.

- `get_area double :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the area of the triangle formed by(if exists) the points c_1,c_2 and c_3.
- `get_perimeter double :(cdouble c1,cdouble c2,cdouble c3)`: Given three complex numbers $c_1 = a_1 + b_1 i$,$c_2 = a_2 + b_2 i$ and $c_3 = a_3 + b_3 i$ this function returns the perimeter of the triangle formed by(if exists) the points c_1,c_2 and c_3.

8 Example programs

8.1 Example program 1:

```
my_centroid cdouble : cdouble c1,cdouble c2,cdouble c3  {
    cdouble centroid;
    centroid = (c1+c2+c3)/3;
    return centroid;
}
main int :  {
    cint a(3,4);
    cint b(5,5),c(-101,100);
    cdouble centroid;
    centroid = my_centroid(a,b,c);
    choice(centroid eq get_centroid(a,b,c)) {
        cprint(centroid);
    }
    default {
        cprint(is_triangle(a,b,c));
    }
    return 0;
}
```

Figure 1: Output for table 'department' and k=10

8.2 Example program 2:

```
main int :  {
    cint a(3,4);
    cint b(5,5),c(-101,100);
    cdouble centroid;
    centriod = get_centroid(a,b,c);
    cprint(centroid);
    circumcente= get_circumcenter(a,b,c);
    cprint(circumcenter);
    orthocenter = get_orthocenter(a,b,c);
    cprint(orthocenter);
    choice (dist(centriod,circumcenter) eq dist(orthocenter,centroid)*2){
        cprint(1); //ratio verified
    }
    default {
        cprint(-1);
    }
    //circum centriod orthocenter
    //      2        1
    return 0;
}
```

Figure 2: Output for table 'department' and k=10