1. 由用户输入一个double类型的值，输出它的平方根。

x的平方根采用如下迭代公式进行计算：

#include <stdio.h>

#define eps 1e-7

int main()

{

    double x;

    printf("Please input a number:");

    scanf("%lf",&x);

    double y = x;

    while (y \* y - x >= eps || y \* y - x <= -eps) {

        y = 1.0/2.0 \* (y + x / y);

    }

    printf("The squre root of %f is %f\n", x, y);

    return 0;

}

1. 不调用系统库函数，自己编写一个函数计算x的正弦值。

x的正弦值采用如下迭代公式进行计算：

其中。

#define eps 1e-9

double Sin(double x){

    double y = 0;

    double delta = x;

    int n = 1;

    while (delta >= eps || delta <= -eps ){

        y += delta;

        delta \*= -x/ (n + 1) \* x /( n + 2);

        n += 2;

    };

    return y;

}

int main()

{

    double x;

    for (x = 0.0; x <= 6.28; x += 0.01){

        printf("Sin(%4.2f)=%9.6f\t", x, Sin(x));

        printf("%9.6f\n", sin(x));

    }

    return 0;

}

3. 编写程序，输入一个整数并输出，输出时要求从右边起每三个数位加一个","。

   程序运行示例：

   12345

   12,345

4. 编写程序，从用户输入的带路径和扩展名的文件名中提取文件名（去掉路径前缀和扩展名）

    程序运行示例：

    abc/def/file.txt

    file

5. 程序改错。以下程序判定一个字符串是否是另一个字符串的逆序，如果是，返回值为1，如果不是，返回值为0。该程序中存在两处错误，请改正。

#include <stdio.h>

#include <string.h>

int is\_reverse(char \*word1, char \*word2) {

    int i = 0;

    int j = strlen(word2);

    while (j > 0) {

        if (word1[i] != word2[j])

            return 0;

        i++;

        j--;

    }

    if (word1[i] != '\0')

        return 0;

    else

        return 1;

}

int main(int argc, char const \*argv[])

{

    printf("%d\n", is\_reverse("stop", "pots"));

    printf("%d\n", is\_reverse("stop", "pot"));

    return 0;

}

6. 检索一个单词表中所有的回文。单词表存放在文件word.txt中，每行存放一个单词。

7. 完成以下的divmod函数，计算整数a除以b所得到的商和余数。

#include <stdio.h>

struct pair {

    int quotient;

    int remainder;

};

struct pair divmod(int a, int b)

{

}

int main()

{

    int a = 8, b = 3;

    struct pair result = divmod(a, b);

    printf("The quotient is %d, and the remainder is %d", result.quotient, result.remainder);

    return 0;

}