

2021-2022 学年第一学期《程序设计基础实践》期末试题

基于自定义有理数库使用 Machin 公式求 π 值

河海大学信息学部计信院 施小华

特别提醒：

1. 请仔细阅读试题要求。
2. 不得传播和公开发布本试题。

一、Machin 公式简介

已知 Machin 公式如下：

$$\text{Machin: } \pi = 16 \arctan \frac{1}{5} - 4 \arctan \frac{1}{239}$$

$$\arctan x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \frac{x^9}{9} \dots$$

可以通过求 $\arctan(x)$ 的前 n 项来求 π 值。比如：

- 1、求 $\arctan(x)$ 的前 1 项，得

$$\arctan(1/5) = (1/5)$$

$$\arctan(1/239) = (1/239)$$

$$\begin{aligned} \text{则 } \pi &= 16 * \arctan(1/5) - 4 * \arctan(1/239) \\ &= (3804/1195) \\ &\approx 3.1 \end{aligned}$$

此时求得的 π 值可精确到小数点后 1 位

- 2、求 $\arctan(x)$ 的前 2 项，得

$$\arctan(1/5) = (74/375)$$

$$\arctan(1/239) = (171362/40955757)$$

$$\begin{aligned} \text{则 } \pi &= 16 * \arctan(1/5) - 4 * \arctan(1/239) \\ &= (5359397032/1706489875) \\ &\approx 3.14 \end{aligned}$$

此时求得的 π 值可精确到小数点后 2 位

- 3、求 $\arctan(x)$ 的前 20 项，得

$$\arctan(1/5) =$$

$$(2398044094221698971925587559419308359564/12148419630337230046279728412628173828125)$$

$$\arctan(1/239) =$$

$$(39971212836972005552188694633099493028856124950568861767362720221$$

6888523662783499395777809192937272628284/9553175615632174014122131
083370709374868761849675670580834071032727194377180749182777183213
6054188624392775)

则 $\pi = 16 * \arctan(1/5) - 4 * \arctan(1/239)$
=

(66171938080125405917769773480472803574432966365690584996721299349
682441765190724033918961782945904768595268952538318114299048395355
2/2106318207884556176009084156991904516539626781647832342078237030
744925312690115194836765297894347314096503396285697817802429199218
75)

$\approx 3.14159265358979323846264338327$

此时求得的 π 值可精确到小数点后 29 位。

二、程序要求

按以下四个步骤编写程序：

1、在实验“非负整数的加减乘法运算”的基础上实现非负整数的除法运算：

```
Unsigned* unsigned_div(const Unsigned*,  
                        const Unsigned*,  
                        Unsigned** rem);
```

其中第一个参数表示被除数，第二个参数表示除数，第三个参数用于返回余数。

见 01_unsigned_div 目录。测试代码见 unsigned_test.c。此步骤要求将 unsigned.c 补充完整。

2、在以上成果的基础上实现大整数的比较以及加减乘除法运算：

```
typedef enum {  
    plus = 1, // 表示非负数  
    minus = -1 // 表示负数  
} Sign;  
  
typedef struct {  
    Sign      sign; // 表示大整数的符号（负数或非负数）  
    Unsigned* val;  // 表示大整数的绝对值的大小  
} BigInt;
```

```
BigInt* big_int_from_ll(long long); // 已实现  
BigInt* big_int_copy(const BigInt*); // 已实现  
void    big_int_free(BigInt*);      // 已实现
```

```
char*    big_int_to_str(const BigInt*); // 已实现  
long long big_int_to_ll(const BigInt*); // 已实现，本程序中用于测试  
int       big_int_cmp(const BigInt*, const BigInt*);
```

```
BigInt* big_int_add(const BigInt*, const BigInt*);
```

```

BigInt* big_int_sub(const BigInt*, const BigInt*);
BigInt* big_int_mul(const BigInt*, const BigInt*);
BigInt* big_int_div(const BigInt*, const BigInt*, BigInt** rem);

```

见 02_big_int 目录。测试代码见 big_int_test.c。此步骤要求将 big_int.c 补充完整。

3、在以上成果的基础上实现有理数的加减乘除法运算：

```

typedef struct {
    BigInt* num;    // 表示有理数的分子
    BigInt* denom;  // 表示有理数的分母
} BigRat;

BigRat* big_rat_from_ll(long long, long long); // 已部分实现
BigRat* big_rat_copy(const BigRat*);          // 已实现
void    big_rat_free(BigRat*);                 // 已实现

// 已实现，本程序中主要用于测试
char* big_rat_to_str(const BigRat*);
void  big_rat_to_ll(const BigRat*, long long* num, long long* denom);

BigRat* big_rat_add(const BigRat*, const BigRat*);
BigRat* big_rat_sub(const BigRat*, const BigRat*);
BigRat* big_rat_mul(const BigRat*, const BigRat*);
BigRat* big_rat_div(const BigRat*, const BigRat*);

```

见 03_rational 目录。测试代码见 rational_test.c。此步骤中加减乘除运算的结果应化简，参见 rational.c 中函数 big_int_cancel 的注释。此步骤要求将 rational.c 补充完整。

4、在以上成果的基础上使用 Machin 公式，且通过求 $\arctan(x)$ 的前 40 项来求 π 值。此时会得到 π 的一个分数表示形式，即一个有理数。要求程序输出此有理数的分子和分母（分子分母应化简，即同时除以它们的最大公约数）、以小数形式表示的 π 的近似值、分子和分母的位数、以及 π 值精确到小数点后的位数。格式如下（以求 $\arctan(x)$ 的前 20 项为例）：

```

numerator   = 6617193808012540591776977348047280357443296636569058499672129934968244176519072403391896178
29459047685952689525383181142990483953552
denominator = 2106318207884556176009084156991904516539626781647832342078237030744925312690115194836765297
89434731409650339628569781780242919921875
 $\pi$          = 3.14159265358979323846264338327

numerator    : 132 digits
denominator  : 132 digits
 $\pi$  decimal places : 29 accurate

```

见 04_pi 目录。此步骤的程序框架见 pi_by_Machin_formula.c，要求将 pi_by_Machin_formula.c 补充完整。

特别注意，在以上 4 步中：

- 1、只允许改动上述要求补充完整的 4 个 .c 文件，不得改动其它文件；
- 2、补充的代码只能出现在文件中标识为“// 在此处补充完整”或者“// 在此处可定义若干辅助函数，以供定义其它函数时使用”处；
- 3、除了 C 语言标准库头文件，不允许 #include 其它头文件（包括第三方库头文件和自定义的头文件）。

各个函数的功能和用法可参见各 .c 文件中的代码和注释。

三、提交内容、提交方式、及截止时间

1、提交内容

- 1) 文件 completed.txt
(里面写明专业、班级、学号、姓名，以及完成了哪几步)
- 2) 各步骤中所有的源代码
(每个步骤放在一个目录，即“2021 试题.zip”中的 4 个目录)
- 3) 各步骤的运行结果
(将运行结果截图并保存为 result.png，和各步骤的源代码放在一起)

2、提交方式

将提交内容压缩为“学号_姓名.zip”文件，其中“学号”替换为自己的学号，“姓名”替换为自己的姓名。每个同学将压缩好的 .zip 文件交给本班学委，重修的同学交给杨阳同学（QQ 号：2933902068），转专业及其它同学交给李忆茹同学（QQ 号：1693205781）。

各班学委再将本班作业压缩为“专业_班级.zip”文件，其中“专业”替换为本班同学的专业，“班级”替换为本班同学的班级；杨阳同学将重修同学的作业压缩为“重修.zip”；李忆茹同学将转专业及其它同学的作业压缩为“转专业及其它.zip”。

各班学委及杨阳同学和李忆茹同学将压缩好的 .zip 文件交给我。

3、截止时间

各位同学应将压缩好的 .zip 文件于 2022 年 1 月 10 日 20:00 前交给相应的收作业的同学。各位收作业的同学应将压缩好的 .zip 文件于 2022 年 1 月 10 日 20:30 前交给我。

本期末作业**要求独立完成，若发现抄袭，抄袭者和被抄袭者期末成绩计 0 分。**
本期末作业**应按时提交，单独发我或者补交一律不收，期末成绩计 0 分。**