C Programming II
2021 Spring
Final

Instructor: Po-Wen Chi

Date: 2020.06.05 14:00-17:00

Policies:

- Offline test unless you complete the exam and want to submit your code.
- Do not forget to include your Makefile. TA will only use the command make to build your program. If make fails, you will get zero points and no room for bargaining. So if you do not know how to solve a problem, please, do not include it in your Makefile.
- I do not care your source code file names, but the executive binary names should be **fin01, fin02, fin03, fin04**.
- You can ask TA if you do not understand the problems.

1 Polynomial Arithmetic (40 pts)

In this class, I have shown you how Linus developed a linked-list. Now I want you to use the Linked List structure developed by Linus to develop a polynomial structure which supports only one variable. An example is as follows:

$$f(x) = 2x^3 + 5x - 4.$$

The implementation is simple. First, you need to design a structure for a polynomial term, where you should record its coefficient and degree. Then, you can use a linked-list to combine all polynomial terms. In this problem, I promise that all coefficients are **32-bits integers** and **all degrees are larger than or equal to zero**.

Please implement the following things:

```
typedef struct _sPolyTerm
{
} sPolyTerm;

void set_poly( struct list_head *pPoly, int coefficient, int degree );
void print_poly( struct list_head *pPoly );
```

```
7  void clear_poly( struct list_head *pPoly );
8  void add_poly( struct list_head *pResult, struct list_head *pPoly1, struct
       list_head *pPoly2 );
9  void del_poly( struct list_head *pResult, struct list_head *pPoly1, struct
       list_head *pPoly2 );
10 void mul_poly( struct list_head *pResult, struct list_head *pPoly1, struct
       list_head *pPoly2 );
11 void pow_poly( struct list_head *pResult, struct list_head *pPoly, int n );
12 int eval_poly( struct list_head *pPoly, int a );
13 void diff_poly( struct list_head *pResult, struct list_head *pPoly );
14 double int_poly( struct list_head *pPoly, int a, int b );
```

- set_poly
  - Set the coefficient of the given order term.
- print_poly
  - Print the polynomial in the degree descending order.
  - Example: print $3x^3 + x + 10$
    3x∧3+x∧1+10.
  - If there is no terms in the polynomial, print **invalid**.
- clear_poly
  - Clear all terms of a polynomial.
  - Do not forget to free the used memory.
- add_poly
  - $f(x) = f_1(x) + f_2(x)$.
- del_poly
  - $f(x) = f_1(x) - f_2(x)$.
- mul_poly
  - $f(x) = f_1(x) * f_2(x)$.
- pow_poly
  - $g(x) = (f(x))^n$.
- eval_poly
  - return $f(a)$.
  - You do not need to consider the overflow issue.
- diff_poly

  - $g(x) = f'(x)$.

- int_poly
  - return $\int_a^b f(x)$.

For your convenience, I will give you an example code called. Moreover, you do not need to check the invalid input. So do not say that you do not have enough time.

The TA will prepare **fin01.c** for you where **poly.h** is included in **fin01.c**. Do not forget to build **fin01.c** in your Makefile.

2 Picture Rotation (30 pts)

Given a BMP picture and a rotation angle, please generate a picture after **clockwise** rotation. Figure 1 is an example. The output file name should be **output.bmp**. You should fill the redundant part with white color.

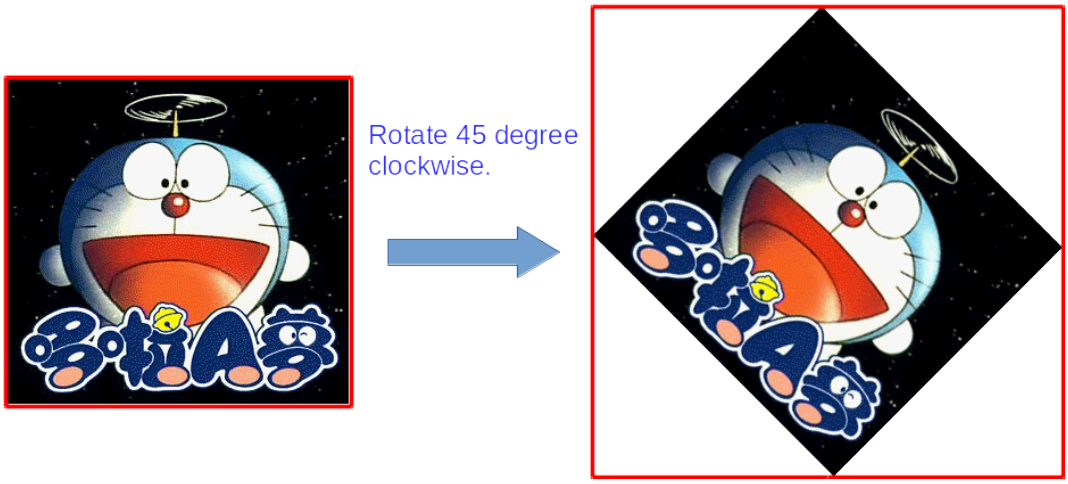

Figure 1: Picture clockwise rotation. The red boundary line is for your reference only. You do not need to draw the boundary.

```
$ ./fin02 -i doraemon.bmp -o output.bmp -a 45
$ ./fin02 -a 45 -o output.bmp -i doraemon.bmp
$ ./fin02 --angle 45 --input doraemon.bmp --output output.bmp
$ ./fin02 --output output.bmp --angle 45 --input doraemon.bmp
$ ./fin02 --help
fin02:
    -a, --angle: angle for clockwise rotation
    -i, --input: input file
    -o, --output: output file
    -h, --help: This description
```

Note that the rotation precision is not a concern in this problem. Do not forget to handle the error cases.

3 Online Dictionary (30 pts)

As you know, my English is poor. So I need you to develop a dictionary for me. The user can input an English word and you should display the word information for the user. Do not worry, I do not ask you to make a dictionary. There are lots of online dictionaries. In this problem, you should use the following dictionary.

`https://simple.wiktionary.org/wiki/`

You can use the site to look up the English word.

Your program behavior should be as follows.

```
$ ./fin03 -w program
Verb: program/programs/programed/programed/programing

1. (transitive & intransitive) When you program a computer, you use code to
   tell the computer what to do.
    EX: The student can program the computer to calculate all of his
   statistics.
    EX: All of the math students know how to program.

Noun: program/programs

1. (countable) A program is a number of steps that are followed to reach a
   goal or end.
    EX: His new exercise program gets more difficult every week.
    EX: Oxford College has an excellent English language program.

2. (countable) A program is a guide to a dramatic or theatrical performance.
    EX: After he entered the theatre, he bought a copy of the program.

3. (countable) A (television or radio) program is a broadcast.
    EX: Star Trek is her favourite program.

4. (countable) A (computer) program is the code that tells a computer what to
   do.
    EX: Microsoft Word is one of the most popular programs in the world.
    EX: The new program has a lot of problems. My computer crashes every time
   I try to run it.
$ ./fin03 -h
fin02:
    -w <word>
    -h
```

You must use **libcurl** to derive the site information. I promise that the TAs have installed **libcurl** in their computers. Do not forget to link the library in your Makefile. All techniques are allowed.

4 Bonus: Your Comments (5 pts)

Again, any comments are welcomed. However, you will get nothing if you leave this question blank.