

Lab 12/03

Do not use Global variable and array in this lab

1. HW problem :

Write a program for an Automatic Teller Machine that dispenses money. The user should enter the amount desired(a multiple of 10 dollars)and the machine dispenses this amount **using the least number of bills**. The bills dispensed are 50s, 20s, and 10s. **Write a function** that determines how many of each kind of bill to dispense.

Let user continuously input the amount until entering ctrl-d or ctrl-z.

Be sure to handle the exception when the amount is not a multiple of 10 dollars.

Example :

```
input amount desired:
390
50: 7
20: 2
10: 0
----input amount desired----:
1999
illegal input
----input amount desired----:
180
50: 3
20: 1
10: 1
----input amount desired----:
^Z
```

2. HW problem : Please complete following three subroutines

A. void order(...)

Arguments : two **double** parameters n1, n2 that are **passed by address**

Function : the smaller of its two argument values is stored in its first actual parameter and the larger is stored in its second actual parameter

B. void sumavg(...)

Arguments : five **double** parameters n1, n2, n3, sump, avgp. Where **n1, n2, n3** are **passed by values**, and **sump, avgp** are **passed by address**.

Function : compute the sum and average of n1, n2 and n3, storing them into sump and avgp

C. void separate(double num, char * sign, int * wholep, double * fracp)

Arguments : as shown above

Function : find the sign, whole number magnitude and fractional parts of the input parameter num. (you may use functions in math.h for help)

Let user continuously input values to test subroutine A, B and C. Each subroutine test stops when user input ctrl-d or ctrl-z, and then followed by next subroutine test.

Example :

```
-----subroutine A-----
1.5 2.5
1.500000, 2.500000
3.5 1
1.000000, 3.500000
^Z
-----subroutine B-----
1.1 -2.2 3.3
2.200000, 0.733333
100 90.5 -100
90.500000, 30.166667
^Z
-----subroutine C-----
-3.14159
- 3 0.141590
12345.6789
+ 12345 0.678900
^Z
```

3.

A. Fibonacci number can be generated by power of matrix $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} =$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n. \text{ For example, } \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^3 =$$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix}, \text{ the series of element } a_{12} \text{ with } n=3 \text{ are Fibonacci series } 1 \ 1 \ 2.$$

Please define the function of matrix multiplication and calculate Fibonacci series with length (n+1).

You can't use other variables in the main function or define other functions, and you must follow the prototype of function void mul(...).

B. **Write a function(s)** that translates the amount of Arabic number into Chinese representation.

Let user continuously input the amount until entering ctrl-d or ctrl-z.

The range of the amount of Arabic number is [0, INT_MAX].

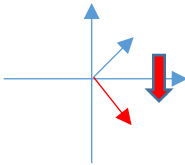
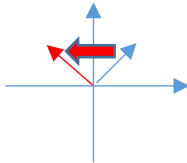
Example:

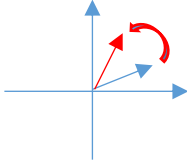
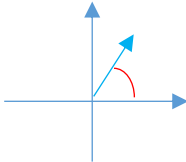
```
----part A----
10
1 1 2 3 5 8 13 21 34 55 89
15
1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
^Z
----part B----
0
零
2147483647
貳拾壹億肆千柒百肆拾捌萬參千陸百肆拾柒
100000123
壹億零壹百貳拾參
1009040300
十億零玖百零肆萬零參百
123456
十貳萬參千肆百伍拾陸
^Z
```

4.

A. Basic vector transformation

Please design following functions to handle transformations of a 2D vector.

Function	Prototype	Description
reflectX	y coordinate of the vector	Reflect the vector over x axis 
reflectY	x coordinate of the vector	Reflect the vector over y axis 

rotate	the vector and an angle input by users	Rotate the vector 'angle' radian clockwise 
angle	the vector and a parameter phi indicating radian of the vector	Calculate and modify phi with the angle of the vector 

Rotation matrix for 2D vector is shown below

$$\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

Let users input an initial vector, then choosing which option to do under a menu driven system. You need to **output the transformed 2D vector and its angle** each time before users choose the exit option.

The angle of a 2D vector (x, y) is defined as:

$$\text{ang}(\text{rad}) = \arctan\left(\frac{y}{x}\right)$$

You may use **atan2** function in math.h for help.

Notice : we will test the rotation function using values in **rad unit.**

B. Encoder

Please write a function with prototype “void encoder(int *x, int N, int bias)” to process the encoding rule showed as following.

Step 1. Right shift the code x

Right shift the code x N times, and the overflow digit will be put to the front.

Example: 12345 right shifts 102 digits ➔ 45123

Step 2. Add the bias

Add the bias to the code x.

Example: $45123+99999 \rightarrow 145122$

Step 3. Truncate the extra digits

If the length of result is longer than original length, the extra digits will be truncated.

Example: $145122 \rightarrow 45122$

The **input is consisted of three integers**, which represent code x, shift digits, and bias. And the program should be **stop until the input is ctrl-d or ctrl-z**.

Example:

```
----part A----
please input initial vector (u,v):1 1
what can I help u?
1:rotate
2:mirrorX
3:mirrorY
4:exit
1
please input degree phi in radius:1.7
(-1.120509,0.862820), 2.485398
what can I help u?
1:rotate
2:mirrorX
3:mirrorY
4:exit
2
(-1.120509,-0.862820), -2.485398
what can I help u?
1:rotate
2:mirrorX
3:mirrorY
4:exit
4
(-1.120509,-0.862820), -2.485398
----part B----
12345 102 99999
45122
^Z
```