PROG24310: Programming Languages

Evaluation: 10 points, 10% of your final grade.
Due date: October 9th, 2016 (11:30pm)

Problem 1 (5 points).

Write a C program fibonacci.c that takes as input an integer number **n** and calculates the Fibonacci number $\mathbf{F_n}$ using recursive function calls.

Your program should have two functions: **main()** and **fibonacci()**.
All the printf's/scanf's are placed in the **main()** function. main's job is to do all handling of user input/output, call **fibonacci** function when appropriate, and report its result to the user.
Inside **main** function it should be called as: `fibonacci(n);`
fibonacci function must return unsigned long value.

Demo. After your program is compiled, the user should be able to run it as follows:

```
Please enter the integer number n: -11
Wrong input!
Please enter the integer number n: 14
Fibonacci number F_14 is 377.
```

Note: use shorthand notation(s) in the **fibonacci** function.

Problem 2 (5 points).

Write a C program dochange.c that takes as input a money amount in cents (variable **money**) and creates an integer array **change[]** of size 5, where the 0th item is the number of pennies, the 1st is the number of nickels, the 2nd is the number of dimes, the 3rd is the number of quarters, and 4th is the number of dollars.

Your program should have two functions: **main()** and **dochange()**.

All the printf's/scanf's are placed in the **main()** function. main's job is to do all handling of user input/output, call **dochange** when appropriate, and report its results to the user.
The prototype of **dochange** function should be as follows: `void dochange(int, int*);`
Inside **main** function it should be called as: `dochange(money,change);`

Note: do not write a program with many nested loops. Instead, see if there is a strategy that always gives you the best result without having to search through all possible combinations.

Demo. After your program is compiled, the user should be able to run it as follows:

```
Please enter the amount of money you would like changed: -5
Wrong amount!
Please enter the amount of money you would like changed: 236
2 Dollar(s)
1 Quarter(s)
1 Dimes
0 Nickel(s)
1 Penny(ies)
```

Requirements:

Each C program must be:

- Reasonable optimized: use function(s), proper data types, minimize the code.
- Check the user input and provide notification if it's wrong (see the demo runs above).

Submission:

- Please make sure your program works in NetBeans/Cygwin!
- Save your programs as text file with extension .txt
- You must upload dochange.c.txt and fibonacci.c.txt files to the Dropbox.
- Your submission must be unique.
- **Late submissions are not accepted!**

Mark Breakdown & Deductions:

Total: 5 points / per program

Missing Functionality	Mark deduction
The program's output is different than the demo run.	-0.25 points
The program is not optimized.	up to -1 point
No input/error checking.	-0.25 points
Different function name/prototype used.	-1 point.
Program doesn't compile/run, or partial functionality is provided.	varies