HW2

How to submit --

Type the following to submit after you finish all the problems:

~lyang11/bin/submit cs304 hw2 test543.c hw2.txt

Problem 1. Command line arguments: test543.c

Let's read the following program (test543.c):

```
---------------------------------
 #include <stdio.h>

 int main (int argc, char *argv[]) {
 int i=0;
 printf("\ncommand line args count=%d\n", argc);

 /* First argument is executable name only */
 printf("executable name=%s\n", argv[0]);

 for (i=1; i< argc; i++) {
    printf("arg%d=%s\n", i, argv[i]);
 }

 return 0;
 }
---------------------------------
```

Copy this problem to test543.c, compile it:
>gcc -o test543 test543.c

Now run the program:
>./test543 abc df jkl

It will print out the following:

```
command line args count=4
executable name=./test543
arg1=abc
arg2=df
arg3=jkl

-------------------------
```

In this example, argc is the number of arguments including the executable name ("./test543"), argv is a pointer to a pointer that points to a char or a string.
In the declaratin of main, there is a declaration of char *argv[]. This means argv is an array whose elements are pointers to type char.

That is, argv[0], argv[1] are all pointers that point to a string.
In our example running, argv[0] contains the starting address of string "./test"; argv[1] contains the starting address of string "abc";
argv[2] contains the starting address of string "df"; argv[3] contains the starting address of string "jkl".

So what is *(argv[3]+1)? It should be 'k', i.e., the second char in "jkl".
What is *(argv[3]+3)? It should be 0 (not symbol '0') because that is the end of the string "jkl"

Your job here is to modify the program so that it will accept a number of integers and it will print out the squares of each accepted integers. **Don't use any other functions.** Again you can refer to Problem 4 in HW1.

> ./test543 12 23 22 345

command line args count=5
executable name=./test543
arg1=12
arg2=23
arg3=22
arg4=345
The square of arg1 is 144
The square of arg2 is 529
The square of arg3 is 484
The square of arg4 is 119025

Problem 2: Floating point hw2.txt

(1) Give the floating point representation of -213.

(2) What floating point numbers are represented by the following bit patterns? Give answer in decimal system, no scientific notation.

Note that we have 32 bits word: S(1bit).Exp(8bits).Sig(23bits). Normally the decimal number is $(-1)^{S}*(1.Sig)*2^{(Exp-127)}$ as we discussed in class. There are some exceptions:

1. If Exp=0FF (i.e., 0b11111111), the number will be NaN (not a number) if the significand (Sig) is not 0;
2. If Exp=0FF (i.e., 0b11111111), the number will be ±infinity if the significand (Sig) is 0 depending on the sign (S).

a) 0x00000000
b) 0xBF800000
c) 0x44802000

save it to a file hw2.txt

Problem 3: Values of i and j hw2.txt

In the following program, for each line, write down the values of i and j.

```
main(){
int i, j, *p, *q;
// The following line means i=5; j=i;
j=(i=5);
// The following line means the logical value of i==4 will be assigned to j. That is,
if i is equal to 4, then j=1; otherwise j=0;
j=(i==4);
j=++i;
j=i--;
p=&i;
q=&j;
*p=3;
*q=*p+1;
i=8;
j=9;
(*p)++;
(*q)++;
q=p;
(*q)--;
return 0;
}
```

Save your answer in hw2.txt