CS 240: Data Structure
Problem Set 1

Due: 02/24/2020

Instructions:
Please follow the instruction given at each problem for submission. The deadline of each assignment is 11:59 pm on the due date, unless otherwise specified. It is important that your code/solution is straight forward, not cumbersome. At the beginning of each cpp file, you must include the following:
1. Program Description: description of the program
2. Author: the person who writes this program
3. Date: the very last date/time the program is modified
4. Function Description: description of each function and its parameters.

First Name:

Last Name:

Score: **/120**

Problem 1 Programming: Validator + Perm: 5 +15 pts

In class, we learned how to write the factorial function and the combinatorial function. Remember that $n! = n \times (n-1) \times \cdots \times 1$ and $C(n,k) = n!/((k!) \times (n-k)!)$. Please write out the permutation function **Perm (int n, int k)** where $Perm(n,k) = n!/k!$. Your program will ask for inputs in the main function. Your program will need to verify first that (1) n is a positive integer and (2) k is some positive integer less or equal to n. Your code in main (or you can write a short checking fucntion) will verify the inputs. If not valid, your main will print out “Invalid inputs” and continue to ask for inputs till valid inputs are obtained. Once the inputs are valid, then your main will call the Perm function to compute then the main will print out the result. Please name this file **perm.cpp**.

Problem 2 Bored Post man: array and bool: 20 pts

Peter the postman became bored one night and, to break the boredom, he carried out the following experiments with a row of mailboxes in the post office. These mailboxes were numbered 1 through 150 and beginning with mailbox 2, he opened the doors of all the even-numbered mailboxes, leaving others closed. Next, beginning at mailbox 3, he went to every third mail box, openning its door if it is closed and closing it if open. Then he repeated this procedure with every fourth maiblox, then 5th and so on. When he finished, he was surprised at the distribution of closed mailboxes. Write a program to simulate this experiment and report which mailboxes are closed at the end of experiment.

Problem 3 Struct + Sorting + Pointer : 80 pts

We simply talked about the structure LinkedList (only one direction, not Double LinkedList). Let us define the structure of the LinkedList as

```
struct node
{int value;
node * next;
};
```

On the course blackboard we have the linklist_nosort.cpp. Please modify the code such that
(a) linklist_nosort_1.cpp that print out is 3 1 7 2 9
(b) linklist_sort.cpp that the print out is 1 2 3 7 9
(c) linklist_sort_1.cpp that the print out is 1 3 7 9 2
(d) linklist_sort_2.cpp has the insert function of the form unction **node* insert(node *p, int k)** where p is the address of the beginning of the LinkedList you have built so far and k is the integer you just read in from the array.