**Introduction to Computer Science – 150005**

**Homework Assignment #11**

**Recursion**

**Comments:**

1. At the end of each question, you need to provide a main program that shows the correctness of the functions that you wrote.
2. Use meaningful variable names
3. Comment each program, including a comment before the main program and each function explaining their purpose and how they work, and a comment at the end of the program containing test runs with input and output.
4. Be careful on code readability and appearance (indentation)
5. Make sure to compute exactly what is requested in each question.
6. In the examples below, output is marked in green, input in yellow.
7. Reminder: submit your own work!

**Important notice: The functions for this assignment must be be done with recursion and not loops. A program that is not recursive will not receive any points.**

1. Write a recursive function that receives a non-negative integer n as input. The function should print n times the letter ‘a’ followed by n times the letter ‘b’ with no spaces. If n is 0, the function prints nothing.  
     
   enter a number:  
   3  
   aaabbb

You may use the following main program:  
  
#include <iostream>

using namespace std;

void printABs(int);

int main(){

int num;

cout << "enter a number:"<<endl;

cin >> num;

printABs(num);

return 0;

}

1. Write a recursive function that receives as input an array and its size. The function should return the **index** of the smallest element in the array. The main program inputs from the user a positive integer representing the size of the array and declares a dynamically allocated memory for the array of that size. Afterwards, it inputs the elements for the array, calls the recursive function, and prints the smallest element in the array.  
     
     
     
   enter a number:

6

enter array values:

2 4 5 1 5 6

the smallest is: 1  
  
You may the following s the main program:  
  
int smallest(int \* , int );

int main(){

int \* array;

int num;

do {

cout << "enter a number:"<<endl;

cin >> num;

if (num<=0)

cout << "ERROR" << endl;

} while (num<=0);

array = new int[num];

cout << "enter array values: ";

for(int i = 0; i < num; i++)

cin >> array[i];

cout << "the smallest is: " << array[smallest (array,num)]<< endl;

return 0;

}

1. Write a recursive function that receives as input an array of integers and its size. The function sorts the array in non-descending order using selection sort. The algorithm works as follows:
   * Find the smallest element in the array
   * Swap this number with the first element in the array
   * Sort the array from the second element until the end using this technique

To find the smallest element each time, you should use the recursive function of the previous question.  
  
You may use the following function to swap elements at index i and j in the array.  
  


The main program inputs from the user a positive integer representing the size of the array and declares a dynamically allocated memory for the array of that size. Afterwards, it inputs the elements for the array, prints the original array, calls the recursive function to sort the array, and again prints the array.

enter a number:

6

enter array values:

2 5 7 5 3 1

before:

2 5 7 5 3 1

after:

1 2 3 5 5 7

You may use the following main program:

void swap(int\* , int, int);

int smallest(int \*, int);

void sort(int \*, int);

int main()

{

int \* arr;

int num;

do {

cout << "enter a number:"<<endl;

cin >> num;

if (num<=0)

cout << "ERROR" << endl;

} while (num<=0);

arr = new int[num];

cout << "enter array values: ";

for(int i = 0; i < num; i++)

cin >> arr[i];

cout << "before: \n";

for(int i= 0; i < num; i++)

cout << arr[i] << " ";

cout << endl;

sort(arr, num);

cout << "after: \n";

for(int i= 0; i < num; i++)

cout << arr[i] << " ";

cout << endl;

return 0;

}

1. Write a recursive function that receives as input an array of integers and its size. The function reverses the elements of the array such that the first element will be in the last position, the second element in the next to last position, and so on.

enter a number:

6

enter array values:

1 2 3 4 5 6

before:

1 2 3 4 5 6

after:

6 5 4 3 2 1

You may use the following main program:

void swap(int\*, int, int);

void reverse(int \*, int);

int main(){

int \* arr;

int num;

cout << "enter a number: ";

cin >> num;

arr = new int[num];

cout << "enter array values: ";

for(int i = 0; i < num; i++)

cin >> arr[i];

cout << "before: \n";

for(int i= 0; i < num; i++)

cout << arr[i] << " ";

cout << endl;

reverse(arr, num);

cout << "after: \n";

for(int i= 0; i < num; i++)

cout << arr[i] << " ";

cout << endl;

return 0;

}

1. N Queens Problem  
   Given a chess board having N×N cells, place N queens on the board in such a way that no queen is attacked by any other queen. A queen is attacked by another queen if they appear on the same row, column, or diagonal.

Write a recursive function that places 8 queens on an empty chess board in such away that no queen is being attacked.