********** ATTENTION: Attendance in lab is mandatory for all students. NO EXCEPTION!!!! **********

Lebanese University
Faculty of Science
Section I

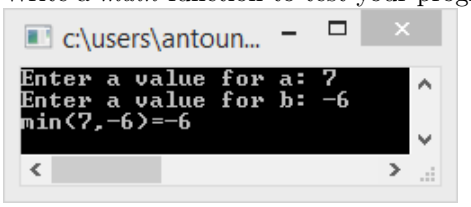
BS - Computer Science
2018-2019

I2206 - INFO 205
Data Structures
LS 1 : Iterative and recursive functions

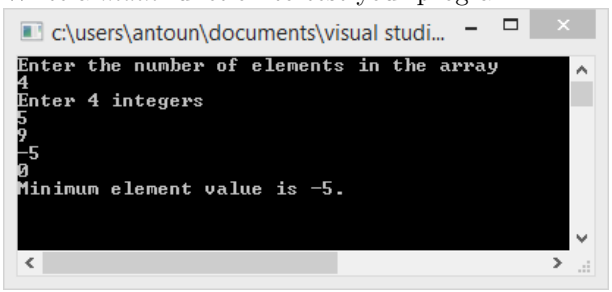
ATTENTION !

You should upload your solution at the latest by Saturday, February 16, 2019 11 p.m. via the web page of the course on www.antoun.me.

Exercise 1 Write a function that returns the minimum of two integers.
Prototype : `int min(int a, int b)`
Write a *main* function to test your program.

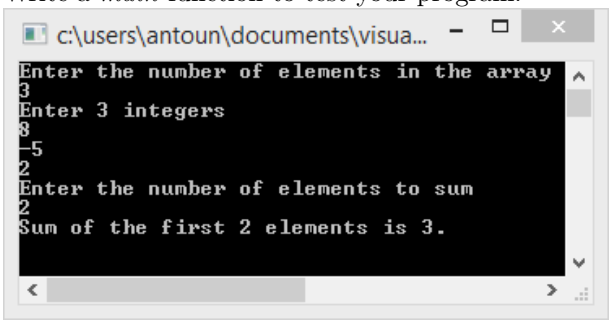


Exercise 2 Using **Exercise 1**, write a function that returns the minimum of an array of integers.
Prototype : `int min(int tab[], int tab_size)`
Write a *main* function to test your program.



Exercise 3 Write a **recursive** function that returns the minimum of an array of integers.
Prototype : `int min(int tab[], int tab_size)`
Write a *main* function to test your program.

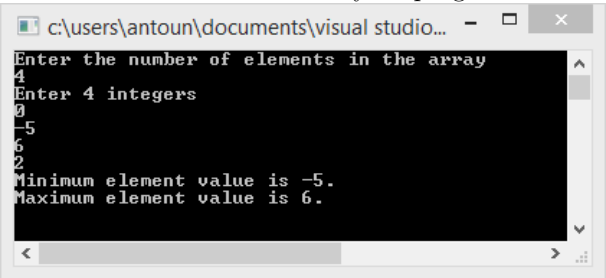
Exercise 4 Write an **iterative** function that returns the sum of the first N integers of an array.
Prototype : `int sum(int tab[], int tab_size, int N)`
Write a *main* function to test your program.



********** ATTENTION: Attendance in lab is mandatory for all students. NO EXCEPTION!!!! **********

Exercise 5 Write a **recursive** function that returns the sum of the first N integers of an array.
Prototype : ...
Write a *main* function to test your program.

Exercise 6 Write an **iterative** function that finds the minimum and the maximum of an array of integers.
Prototype : ...
Write a *main* function to test your program.



Exercise 7 Write a **recursive** function that finds the minimum and the maximum of an array of integers.
Prototype : ...
Write a *main* function to test your program.

For exercises 8 – 11, use the `main` function on page 3

Exercise 8 Write a **recursive** function that replaces all the occurrences of an integer I1 in an array of integers Tab by some another integer I2.

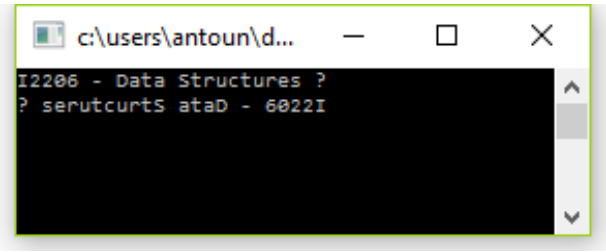
Exercise 9 Write a **recursive** function that computes the set intersection of two integer arrays.

Exercise 10 Write a **recursive** function that determines whether all the elements of an integer array T1 belong to another integer array T2.

Exercise 11 Write a **recursive** function that determines whether an integer array T1 is included in block in another array of integers T2. It is advisable to write another recursive function that determines whether an integer array T1 begins (is in block at the beginning of) another integer array T2.

Exercise 12 Write a recursive function `void mirror(void)` that reads character by character (using the function `getchar()`) a string terminated by ? and then displays the string in reverse order.

```
void main()
{
    mirror();
}
```



```
int main()
{
   int T1[10]={1,3,2,6,4,3,9,11,2,20},
       T2[8]={3,6,10,-3,2,7,9,12},
       T3[4]={-3,3,9,6},
       T4[3]={3,6,10},
       T5[15],
       dim;

   replace(T1,10,3,5); /* replaces 3 by 5 in T1 */
   display(T1,10);

   intersection(T1,10,T2,8,T5,&dim);
   display(T5,dim);

   if (subset(T3,4,T2,8))
       printf("T3 included in T2 \n");
   else
       printf("T3 not included in T2 \n");
   if (sub_array(T3,4,T2,8))
       printf("T3 included in block in T2 \n");
   else
       printf("T3 not included in block in T2 \n");
   if (sub_array(T4,3,T2,8))
       printf("T4 included in block in T2 \n");
   else
       printf("T4 not included in block in T2 \n");

   getch();
   return 0;
}
```