Homework 3 Due:15/05/2023

Q1 A data type called `Node` is used to encapsulate data and to implement the data type `Stack`. Stack is used with a Last-In-First-Out data exchange strategy. This data type is going to be implemented like the Linked-List. Implement the following functions,

- [10p]`void print(Stack* s)`: prints the stack (only for debug purposes).
- [20p]`void push(Stack* s,int val)`: adds the new value to the stack.
- [20p]`int pop(Stack* s)`: removes the last added value from the stack.

in a single file. For more information about stack you may refer to here:

Q2 The data type created in the previous question called `Stack` is going to be used to develop an interpreter. This interpreter uses a special syntax where if two numbers are given consecutively they are pushed into the stack, if an operator is detected then two numbers are popped from the stack, the operator is applied to the numbers and the result is pushed back into the stack. Using this, it is possible to implement a mathematical interpreter.

- [10p]`int str2num(char* str)`: a helper function which converts an entire string into an integer.
- [40p]`int interpret(Stack* s,const char* str)`: uses the provided stack and the string, interprets the string's mathematical syntax content and returns the appropriate result as an integer.

Only integers are used and all operations are integer operations. Only $+, -, *$ and / are supported.

- The entire homework can only use `printf` from `<stdio.h>` and `malloc`, `free` from `stdlib.h`. Any other built-in function or library is not allowed.
- All string manipulations are going to be coded from scratch. Only `str2num` is allowed to be a function.
- Submit a single `*.c` file to NINOVA. Other file types will not be accepted nor graded.
- The given main function is not going to be submitted, only the necessary implementation needs to be submitted.
- Your submission will be compiled with a tester `main.c` file. Your code needs to compile without error, or your grade will be zero.
- Each functionality will be tested and added to your grade.
- Late submissions will be deduced 10p for each day late.
- Cheating is not allowed, once cheating is detected all involved submissions will be graded zero.

```c
//**************************************
//** DO NOT SUBMIT THIS FILE
//**************************************
#include <stdio.h>
//**************************************
typedef struct sNode Node;
typedef struct sStack
{
    Node* head;
} Stack;
void push(Stack* s,int val);
int pop(Stack* s);
void print(Stack* s);
//**************************************
int str2num(char* str);
int interpret(Stack* s,const char* str);
//**************************************
Stack stack={.head=NULL};
int main()
{
    const char* str="1 5 + 6 3 - / 7 *";
    // const char* str=" 10 20 + 2 * 15 - 2 1 * /";
    int result=interpret(&stack,str);
    printf("Result:%d\n",result);

    return 0;
}
```

deliverables/main.c

```c
//**************************************
//** SUBMIT ONLY THIS FILE
//** ONLY PUT THE IMPLEMENTATION
//**************************************
#include <stdio.h>
#include <stdlib.h>
typedef struct sNode
{
    struct sNode* next;
    int val;
} Node;
typedef struct sStack
{
    Node* head;
} Stack;
void push(Stack* s,int val)
{

}

int pop(Stack* s)
{

}

void print(Stack* s)
{

}
//**************************************
int str2num(char* str)
{

}

int interpret(Stack* stack,const char* str)
{

}
```

deliverables/student.c