Nome: João Paulo de Oliveira 11611BCC046

11° Aula prática

Uberlândia

2016

1.Código fonte:

```
#include <stdio.h>
#include <stdlib.h>
#include "list.h"
#include "stack.h"
#include <string.h>
static void print_stack(const Stack *stack) {

        ListElmt        *element;

        int             *data, size,i;
        fprintf(stdout, "Stack size is %d\n", size = stack_size(stack));
        i = 0;
        element = list_head(stack);
        while (i < size) {

          data = list_data(element);
          fprintf(stdout, "stack[%03d]=%03d\n", i, *data);
          element = list_next(element);
          i++;

        }
        return;
}

int Prioridade(char c, char t){
   int pc,pt;
```

```
    if(c == '*' || c == '/')
        pc = 2;
    if(c == '+' || c == '-')
        pc = 1;
    if(c == '(')
        pc = 4;
    if(t == '*' || t == '/')
        pt = 2;
    if(t == '+' || t == '-')
        pt = 1;
    if(t == '(')
        pt = 0;
    return (pc > pt);
}

int main(int argc, char **argv) {

            Stack stack;
            int i,j=0;
            stack_init(&stack, free);
            printf("Digite a operacao para transformar em posfixa:\n");
            char op[100],posop[100], *a;;
            setbuf(stdin,NULL);
    gets(op);
    char* data = (char *)malloc(sizeof(char));
            for (i = 0; i < strlen(op); i++) {
        if(op[i]>=40&&op[i]<=57&&op[i]!=44&&op[i]!=46){
```

```
if(op[i]>=48&&op[i]<=57&&op[i]){
    posop[j]=op[i];
    j++;
}
if(op[i]=='+'||op[i]=='*'||op[i]=='-'||op[i]=='/'){
    while(1){
        a = stack_peek(&stack);
        if((a = stack_peek(&stack))==NULL){
            stack_push(&stack, &(op[i]));
            break;
        }else{
            if(Prioridade(op[i],a[0])){
                stack_push(&stack, &(op[i]));
                break;
            }
            else{
                posop[j]=*((char*)(stack.head->data));
                stack_pop(&stack,(void*)&data);
                j++;
            }
        }

    }
}
if(op[i]=='(')
    stack_push(&stack, &(op[i]));
```

```
            if(op[i] == ')'){
                while(*(a = stack_peek(&stack))!='('){
                    if(op[i] != '('){
                        posop[j]=*((char*)(stack.head->data));
                        j++;
                        stack_pop(&stack,(void*)&data);
                    }
                }
                stack_pop(&stack,(void*)&data);
            }
        }else{
            puts("operacao incorreta, digite uma operacao valida.");
            return -1;
        }
            }
    while((a = stack_peek(&stack))!=NULL){
        posop[j]=*a;
        j++;
        stack_pop(&stack,(void*)&data);
    }
    posop[j]='\0';
    stack_destroy(&stack);
            printf("%s\n",posop);
            return 0;
}
```

2.Print do funcionamento: