

Exercício 1:

```
Ex1.c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include "pilha.h"
4
5 int main(){
6     int quantidade=0, cont=1, numero=0;
7
8     printf("Informe a quantidade de numeros que serao inseridos: ");
9     scanf("%i", &quantidade);
10
11     Pilha pilhaA = pilha(quantidade);
12     Pilha pilhaB = pilha(quantidade);
13
14     while(quantidade >= cont){
15         printf("Digite o %io numero: ", cont);
16         scanf("%i", &numero);
17         empilha(numero, pilhaA);
18         cont++;
19     }
20
21     while (!vaziap(pilhaA)) {
22         numero = desempilha(pilhaA);
23     }
24
25     while (!vaziap(pilhaB) && topo(pilhaB) > numero) {
26         empilha(desempilha(pilhaB), pilhaA);
27     }
28
29     empilha(numero, pilhaB);
30 }
31
32 while (!vaziap(pilhaB)) {
33     empilha(desempilha(pilhaB), pilhaA);
34 }
35
36 printf("Numeros ordenados: ");
37 while (!vaziap(pilhaA)) {
38     printf("%d ", desempilha(pilhaA));
39 }
40
41 destroi(pilhaA);
42 destroi(pilhaB);
43 }
```

```
C:\Users\cloud\OneDrive - Fat X + v
Informe a quantidade de numeros que serao inseridos: 8
Digite o 1o numero: 8
Digite o 2o numero: 1
Digite o 3o numero: 4
Digite o 4o numero: 9
Digite o 5o numero: 2
Digite o 6o numero: 7
Digite o 7o numero: 6
Digite o 8o numero: 5
Numeros ordenados: 1 2 4 5 6 7 8 9
-----
Process exited after 28.1 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

Exercício 2:

```
Ex1.c Ex2.c pilha.h
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include "pilha.h"
4
5 int main(){
6     int quantidade=0, cont=1, numero=0;
7
8     printf("Informe a quantidade de numeros que serao inseridos: ");
9     scanf("%i", &quantidade);
10
11     Pilha pilhaA = pilha(quantidade);
12     Pilha pilhaB = pilha(quantidade);
13
14     while(quantidade >= cont){
15         printf("Digite o %io numero: ", cont);
16         scanf("%i", &numero);
17         empilha(numero, pilhaA);
18         cont++;
19     }
20
21     while (!vaziap(pilhaA)) {
22         numero = desempilha(pilhaA);
23     }
24
25     while (!vaziap(pilhaB) && topo(pilhaB) < numero) {
26         if(topo(pilhaB)==numero){
27             desempilha(pilhaB);
28         }
29         empilha(desempilha(pilhaB), pilhaA);
30     }
31
32     empilha(numero, pilhaB);
33
34     while (!vaziap(pilhaB)) {
35         empilha(desempilha(pilhaB), pilhaA);
36     }
37
38     printf("Numeros em ordem decrescente e sem repeticoes: ");
39     int ultimoNumero = -1;
40     while (!vaziap(pilhaA)) {
41         int numAtual = desempilha(pilhaA);
42         if (numAtual != ultimoNumero) {
43             printf("%d ", numAtual);
44             ultimoNumero = numAtual;
45         }
46     }
47
48     destroi(pilhaA);
49     destroi(pilhaB);
50 }
```

```
C:\Users\cloud\OneDrive - Fat X + v
Informe a quantidade de numeros que serao inseridos: 12
Digite o 1o numero: 5
Digite o 2o numero: 2
Digite o 3o numero: 1
Digite o 4o numero: 7
Digite o 5o numero: 2
Digite o 6o numero: 6
Digite o 7o numero: 1
Digite o 8o numero: 9
Digite o 9o numero: 2
Digite o 10o numero: 7
Digite o 11o numero: 8
Digite o 12o numero: 3
Numeros em ordem decrescente e sem repeticoes: 9 8 7 6 5 3 2 1
-----
Process exited after 27.23 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

Exercício 3:

```
Ex1.c Ex2.c Ex3.c Exercico_3[1].c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include "pilha.h"
5
6 void inverterPalavras(char *frase) {
7     Pilha pilhaLetras = pilha(strlen(frase));
8
9     int i = 0;
10    while (frase[i] != '\0') {
11        if (frase[i] == ' ') {
12            while (!vaziap(pilhaLetras)) {
13                printf("%c", desempilha(pilhaLetras));
14            }
15            printf(" ");
16        } else {
17            empilha(frase[i], pilhaLetras);
18        }
19        i++;
20    }
21
22    while (!vaziap(pilhaLetras)) {
23        printf("%c", desempilha(pilhaLetras));
24    }
25
26    destroip(&pilhaLetras);
27 }
28
29 int main() {
30     char frase[100];
31
32     printf("Digite uma frase: ");
33     fgets(frase, 100, stdin);
34
35     printf("Frase com as letras invertidas:\n");
36     inverterPalavras(frase);
37
38     return 0;
39 }
```

```
C:\Users\cloud\OneDrive - Fat X + v
Digite uma frase: Claudio Henrique Pinheiro de Oliveira
Frase com as letras invertidas:
oidualC euqirneH oriehniP ed
arievilO
-----
Process exited after 9.611 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

Exercício 4:

```
Ex1.c Ex2.c Ex3.c Ex4.c Exercico_4[1].c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include "pilha.h"
5
6 int expressaoBalanceada(const char *expressao) {
7     Pilha P = pilha(strlen(expressao));
8
9     for (int i = 0; expressao[i] != '\0'; i++) {
10        if (expressao[i] == '{' || expressao[i] == '[' || expressao[i] == '(') {
11            empilha(expressao[i], P);
12        } else if (expressao[i] == '}' || expressao[i] == ']' || expressao[i] == ')') {
13            if (vaziap(P)) {
14                destroip(&P);
15                return 0;
16            }
17
18            char abertura = desempilha(P);
19            if ((expressao[i] == '}' && abertura != '(') ||
20                (expressao[i] == ']' && abertura != '[') ||
21                (expressao[i] == ')' && abertura != '{')) {
22                destroip(&P);
23                return 0;
24            }
25        }
26    }
27
28    int balanceada = vaziap(P);
29    destroip(&P);
30    return balanceada;
31 }
32
33 int main() {
34     char expressao[100];
35
36     printf("Digite uma expressao com chaves, colchetes e parenteses: ");
37     fgets(expressao, 100, stdin);
38
39     expressao[strlen(expressao) - 1] = '\0';
40
41     if (expressaoBalanceada(expressao)) {
42         printf("A expressao \"%s\" esta balanceada.\n", expressao);
43     } else {
44         printf("A expressao \"%s\" nao esta balanceada.\n", expressao);
45     }
46 }
```

```
C:\Users\cloud\OneDrive - Fat X + v
Digite uma expressao com chaves, colchetes e parenteses: ([{}])
A expressao "([{}])" esta balanceada.
-----
Process exited after 21.72 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

```
Ex1.c Ex2.c Ex3.c Ex4.c Exercico_4[1].c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include "pilha.h"
5
6 int expressaoBalanceada(const char *expressao) {
7     Pilha P = pilha(strlen(expressao));
8
9     for (int i = 0; expressao[i] != '\0'; i++) {
10         if (expressao[i] == '{' || expressao[i] == '[' || expressao[i] == '(') {
11             empilha(expressao[i], P);
12         } else if (expressao[i] == '}' || expressao[i] == ']' || expressao[i] == ')') {
13             if (vaziap(P)) {
14                 destroip(&P);
15                 return 0;
16             }
17
18             char abertura = desempilha(P);
19             if ((expressao[i] == '}' && abertura != '{') ||
20                 (expressao[i] == ']' && abertura != '[') ||
21                 (expressao[i] == ')' && abertura != '(')) {
22                 destroip(&P);
23                 return 0;
24             }
25         }
26     }
27
28     int balanceada = vaziap(P);
29     destroip(&P);
30     return balanceada;
31 }
32
33 int main() {
34     char expressao[100];
35
36     printf("Digite uma expressao com chaves, colchetes e parenteses: ");
37     fgets(expressao, 100, stdin);
38
39     expressao[strcspn(expressao, "\n")] = '\0';
40
41     if (expressaoBalanceada(expressao)) {
42         printf("A expressao \"%s\" esta balanceada.\n", expressao);
43     } else {
44         printf("A expressao \"%s\" nao esta balanceada.\n", expressao);
45     }
46 }
```

```
C:\Users\cloud\OneDrive - Fat X + v
Digite uma expressao com chaves, colchetes e parenteses: ([{}]{})
A expressao "([{}]{})" nao esta balanceada.

-----
Process exited after 9.349 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

Exercício 5:

```
Ex6.c Ex5.c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include "pilha.h"
4
5 int main(void){
6     Pilha P = pilha(5);
7     char s[11];
8     for(int i=1;i<=3;i++){
9         printf("? ");
10        gets(s);
11        empilha(s,P);
12    }
13    while( !vaziap(P) ) puts(desempilha(P));
14    return 0;
15 }
```

Registro do Compilador Depurador Resultados da Busca Fechar

Mensagem

ud\OneDrive - Fatec Centro Paula Souza\... In function 'main':
ud\OneDrive - Fatec Centro Paula Souza\3*... [Warning] passing argument 1 of 'empilha' makes integer from pointer without a cast

```
C:\Users\cloud\OneDrive - Fat X + v
? um
? dois
? tres
tres
tres
tres

-----
Process exited after 8.562 seconds with return value 0
Pressione qualquer tecla para continuar. . . |
```

Está retornando 3 vezes o nome “três”, pois cada inserção nova de valores na pilha ela é substituída, ficando somente a última.

Exercício 6:

```
Ex6.c
1  #include <stdio.h>
2  #include <string.h>
3  #include "pilha.h"
4
5  int main(void) {
6      Pilha P = pilha(5);
7      char s[11];
8      for(int i=1; i<=3; i++) {
9          printf("? ");
10         gets(s);
11         empilha(strdup(s),P);
12     }
13     while( !vaziap(P) ) puts(desempilha(P));
14     destroip (P);
15     return 0;
16 }
```

Registro do Compilador Depurador Resultados da Busca Fechar

Filename: C:\Users\claud\OneDrive - Fatec Centro Paula Souza\3º Semestre\Estrut
Processing makefile...

```
C:\Users\claud\OneDrive - Fat x + v
? um
? dois
? tres
tres
dois
um

-----
Process exited after 6.031 seconds with return value 3221225477
Pressione qualquer tecla para continuar. . . |
```