

Aluno: Lucas Victor Dias

Professor: Roberto Fontes

Curso: Licenciatura em Computação

Disciplina: Estrutura de dados

Atividade do Capítulo 2

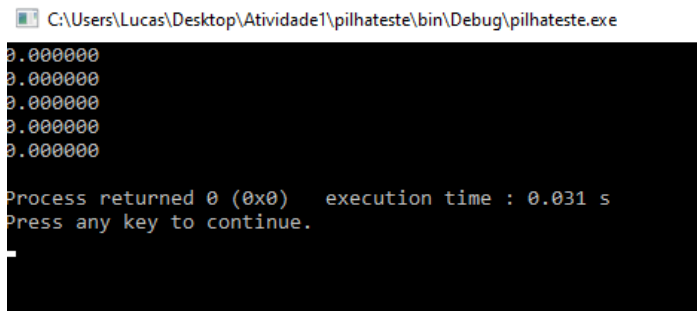
Atividade 2.1)

Resposta marcada em vermelho e amarelo.

```
#include <stdio.h>
#include "../ed/pilha.h" // pilha de int
int main(void) {
    Pilha P = pilha(3);
    empilha(1,P); //1 esta no topo
    empilha(2,P); //2 esta no topo
    printf("%d\n",desempilha(P)); // Retorna o valor 2, o qual foi retirado da pilha
    printf("%d\n",desempilha(P)); // Retorna o valor 1, o qual foi retirado da pilha e
    ela esta VAZIA.
    printf("%d\n",desempilha(P)); // Como a pilha esta vazia, sera retornado o erro
    stack underflow (ERRO DE PILHA VAZIA) ou seja, o PROGRAMA TRAVA
    return 0;
}
```

Atividade 2.2)

//Será exibido 5 linhas com o numero 0, como na imagem abaixo.



```
C:\Users\Lucas\Desktop\Atividade1\pilhateste\bin\Debug\pilhateste.exe
0.000000
0.000000
0.000000
0.000000
0.000000
Process returned 0 (0x0) execution time : 0.031 s
Press any key to continue.
```

//Isso se explica pois, ele desempilha com sucesso e exibe o retorno “0” da função desempilha.

Atividade 2.3)

//Programa que inverte frases (ARQUIVO .C EM ANEXO)

```
#include <stdio.h>
#include <stdlib.h>

typedef struct plv {
    char conteudo;
    struct plv *prox;
} palavra;

void empilha(char c, palavra *topo) {
    palavra *nova = malloc(sizeof(palavra));
    nova->conteudo = c;
    nova->prox = topo->prox;
    topo->prox = nova;
}

char desempilha(palavra *topo) {
    palavra *p = topo->prox;
    char c = p->conteudo;
    topo->prox = p->prox;
    free(p);
    return c;
}

int main() {
    char frase[50];
    palavra cabeca;
    palavra *topo = &cabeca;
    topo->prox = NULL;
    printf("Digite a palavra para ser invertida: /n");
    fgets(frase, 50, stdin);
    for (int i = 0; frase[i] != '\0'; i++) {
        empilha(frase[i], topo);
    }
    printf("A palavra invertida: /n: ");
    while (topo->prox != NULL) {
        putchar(desempilha(topo));
    }

    return 0;
}
```

2.6)

```
int main(void){  
  
    Pilha P = pilha(5);  
    char s[11];  
  
    for(int i=1; i<=3; i++){  
        printf("? ");  
        gets(s);  
        empilha(s,P);  
    }  
  
    while( !vaziap(P) )  
        puts(desempilha(P));  
    return 0;  
  
}
```

Justificativa:

//A saída do programa será:

```
“  
Três  
Três  
Três  
“
```

Quando debugamos o programa, é possível observar que, o nome do número digitado NÃO É enviado direto para função, fazendo um LOOP INUTIL.

Quando o programa entra no laço While para exibir o retorno da função “PUTS”, observamos que, ele efetua todo o laço da função e exibe o último dado que foi inserido no teclado da função MAIN, que foi a palavra “Três”.

Ou seja, tanto a passagem das palavras para a função “empilha()” quanto retorno da função “desempilha()” não estão funcionando corretamente.