

Breno Kazuo Toyonaga dos Santos

RA: 1680482111006

João Pedro Sassi Granado

RA:1680482111012

ADS MA3

Lista I

Exercício 01:

```
1 void occurrences(char *p)
2 {
3     //Calculando o tamanho da string
4     int sizePhrase = 0;
5     for (int i = 0; p[i] != '\0'; i++)
6         sizePhrase++;
7
8     //Criando um vetor de chars auxiliar pra poder fazer as operações
9     char aux[sizePhrase];
10    for (int i = 0; i < sizePhrase; i++)
11        aux[i] = p[i];
12
13    for(int i = 0; i < sizePhrase; i++)
14    {
15        char letter = p[i];
16        int amount = 1; //Amount começa com 1 pra poder já contar a letra 1 vez
17
18        //Verificando se a letra já não foi capturada
19        if(aux[i] != '-')
20        {
21            for (int j = i+1; j < sizePhrase; j++)
22            {
23                if (letter == p[j])
24                {
25                    //Aumentando a quantidade de vezes e alterando a outra letra para -
26                    amount++;
27                    aux[j] = '-';
28                }
29            }
30        }
31
32        if(aux[i] != '-')
33            printf("The letter %c appeared %d times\n", letter, amount);
34    }
35 }
```

Exercício 02:

```
1 Stack *divide(Stack *st)
2 {
3     //Criando uma stack auxiliar pra poder fazer as operações
4     Stack *aux = createStack();
5     while(!emptyStack(st))
6     {
7         push(aux, pop(st));
8     }
9
10    Stack *half = createStack();
11    while(!emptyStack(aux))
12    {
13        //Voltando os valores para a stack original E os valores / 2 na nova
14        float value = pop(aux);
15        push(st, value);
16        push(half, value/2);
17    }
18
19    //Liberando memória
20    freeStack(aux);
21
22    return half;
23 }
```

Exercício 03:

```
1 Queue *queueIntercale(Queue *st1, Queue *st2)
2 {
3     float value = 0;
4     int sizeQueue1 = 0, sizeQueue2 = 0;
5
6     //Criando uma Queue auxiliar e populando ela com os valores da primeira fila
7     Queue *aux1 = createQueue();
8     while(!emptyQueue(st1))
9     {
10         enqueue(aux1, dequeue(st1));
11         sizeQueue1++;
12     }
13
14     //Criando uma Queue auxiliar e populando ela com os valores da segunda fila
15     Queue *aux2 = createQueue();
16     while(!emptyQueue(st2))
17     {
18         enqueue(aux2, dequeue(st2));
19         sizeQueue2++;
20     }
21
22     Queue *final = createQueue();
23     while((sizeQueue1 + sizeQueue2) > 0) //Verificando se ainda tem dados pra serem enfileirados
24     {
25         //Verificando se ainda tem dados na primeira fila
26         if (sizeQueue1 != 0)
27         {
28             value = dequeue(aux1);
29             enqueue(final, value);
30             enqueue(st1, value);
31             sizeQueue1--;
32         }
33
34         //Verificando se ainda tem dados na segunda fila
35         if (sizeQueue2 != 0)
36         {
37             value = dequeue(aux2);
38             enqueue(final, value);
39             enqueue(st2, value);
40             sizeQueue2--;
41         }
42     }
43     return final;
44 }
```