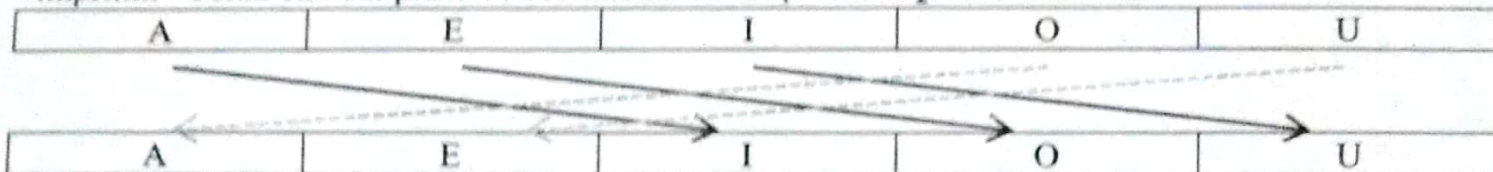


1) (Valor = 2,0) Crie uma versão recursiva da busca binária. Esta função deve receber como parâmetro pelo menos um vetor e a informação a ser pesquisada.

2) (Valor = 2,0) A Cifra de César é uma das mais simples e conhecidas técnicas de criptografia, na qual cada letra é substituída pela correspondente à sua posição mais um número fixo de vezes. Por exemplo, A seria substituído por D, e B por E quando o número for 3. Implemente uma função que receba uma string e substitua apenas as vogais deslocadas por um número inteiro (também passado como parâmetro para a função). Por exemplo, `cifra("Tenha uma boa prova", 2)` deve imprimir "Tonhi emi bai pravi". Pode utilizar as funções `strpos` e `strtam`



3) (Valor = 2,0) Descubra quais as saídas da função quando recebe os valores 3 e 5:

```
função collatz(inteiro: n) {
    escreva("\n", n);
    se (n = 1)
        retorne;
    senão se (n mod 2 = 0)
        collatz(n/2);
    senão
        collatz(3*n + 1);
}
```

4) (Valor = 2,0) Descubra e corrija o erro:

a. O programa a seguir deve imprimir a pirâmide de Pascal (1 erro):

```
função fat(inteiro: n) {
    inteiro: c, res;
    res <- 1;
    para (c <- 1; c <= n; c++) {
        res <- res * c;
    }
    retorne res;
}

início {
    inteiro: i, n, c;
    n <- 5;
    para (i <- 0; i < n; i++) {
        para (c <- 0; c <= n-i; c++)
            escreva(" ");
        para (c <- 0; c <= i; c++)
            escreva(fat(i)/(fat(c) * fat(i-c)));
        escreva("\n");
    }
}
```

Handwritten notes for Pascal's Triangle:

- $n=0, res=1$
- $n=1, res=1$
- $n=2, res=2$
- $n=3, res=6$

Handwritten correction for the Pascal's Triangle code:

```
escreva(fat(i)/fat(c) * fat(i-c));
```

Handwritten Pascal's Triangle:

```

      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
  
```

b. O algoritmo a seguir deve imprimir o MDC de dois números (1 erro):

```
função mdc(inteiro: x, y) {
    se (x = 0)
        retorne y;
    enquanto (y <= 0) {
        se (x > y)
            x <- x + y;
        senão
            y <- y - x;
    }
    retorne x;
}
```

Handwritten correction for the MDC algorithm:

```

enquanto (y <= 0) {
    se (x > y)
        x <- x + y;
    senão
        y <- y - x;
}

```

Handwritten note: $x \rightarrow y$