

# Exercícios Teóricos – Unidade 00

Catarina F. M. Castro – 803531

5)

```
boolean doidao (char c){
    boolean resp= false;
    int v = (int) c;
    if (v == 65 || v == 69 || v == 73 || v == 79 || v == 85 || v == 97 || v == 101 || v == 105 ||
        v == 111 || v == 117){
        resp = true;
    }
    return resp;
}
```

O código checa se o caractere recebido é uma vogal ou não, por meio da sua comparação com os valores decimais da tabela ASCII.

6)

```
boolean isConsoante(String s, int n){
    boolean resp= true;
    if (n!=s.length()){
        if (s.charAt(n)<'0' || s.charAt(n)>'9'){
            if (s.charAt(n)=='A' || s.charAt(n)=='E' || s.charAt(n)=='I' || s.charAt(n)=='O' ||
                s.charAt(n)=='U' || s.charAt(n)=='a' || s.charAt(n)=='e' || s.charAt(n)=='i' ||
                s.charAt(n)=='o' || s.charAt(n)=='u'){
                resp= false;
            } else{
                n++;
                resp=isConsoante(s, n);
            }
        } else {
            resp=false;
        }
    }
    return resp;
}
```

```
boolean isConsoante(String s, int n){
    boolean resp= true;
    if (n != s.length()){
        if (s.charAt(n)<'0' || s.charAt(n)>'9'){
            if (isVogal(s.charAt(n)) == true){
                resp= false;
            } else {
                resp=isConsoante(s, n + 1);
            }
        } else {
            resp=false;
        }
    }
    return resp;
}
```

7)

- Um aluno desenvolveu o código abaixo, corrija-o:

```
boolean isConsoante(String s, int i){
    boolean resp= true;

    if (i == s.length()){
        resp = true;
    } else if (isConsoante(s.charAt(i)) == false){
        resp = false;
    } else {
        resp = isConsoante(s, i + 1);
    }

    return resp;
}
```

```
boolean isConsoante(String s, int i){
    boolean resp= true;
    if (i < s.length()){
        if (!isConsoante(s.charAt(i))){
            resp = false;
        } else {
            resp = isConsoante(s, i + 1);
        }
    } else {
        resp = true;
    }
    return resp;
}
```

8)

- Qual é a diferença entre os dois métodos abaixo?

```
int m1(int i){
    return i--;
}

int m2(int i){
    return --i;
}
```

O primeiro método retorna o valor de i e depois decrementa, enquanto o segundo método decrementa o valor de i e depois o decrementa,

9)

- O que o programa abaixo mostra na tela?

```
byte b = 0; short s = 0; int i = 0; long l = 0;

while (true){
    b++; s++; i++; l++;
    System.out.println(b + " " + s + " " + i + " " + l);
}
```

O programa irá mostrar na tela:

1 1 1 1

2 2 2 2

3 3 3 3

....

Até que se atinja os valores limites de cada tipo primitivo.

10)

- Por que o código abaixo imprime [46 - 11]?

```
int x = 23, y = 23;  
x = x << 1;  
y = y >> 1;  
System.out.println("[ " + x + " - " + y + " ]");
```

O código abaixo imprime esses valores por os operadores << e >> deslocam o valor para a direita (dobrando o valor) e para a esquerda (dividindo o valor por 2), respectivamente.