

# Algoritmia e Programação Procedimental em Java

## Propostas de resolução

### Ficha 3

#### Exercício 1

a)

ED:

num, algarismo, soma INTEIRO

INICIO

```
soma <- 0
REPETE
    LER (num)
    ENQUANTO (num <= 0)

    ENQUANTO (num <> 0)
        algarismo <- num MOD 10
        SE (algarismo MOD 2 = 0) ENTÃO
            soma <- soma + algarismo
        FIMSE
        num <- num DIV 10
    FIMENQUANTO
    ESCRIVER ("A soma dos algarismos pares do número é: ", soma)
```

FIM

b)

ED:

num, algarismo, soma INTEIRO

INICIO

```
LER (num)
ENQUANTO (num>0)
    soma <- 0
    ENQUANTO (num <> 0)
        algarismo <- num MOD 10
        SE (algarismo MOD 2 = 0) ENTÃO
            soma <- soma + algarismo
        FIMSE
        num <- num DIV 10
    FIMENQUANTO
    ESCRIVER ("A soma dos algarismos pares do número é: ", soma)
    LER (num)
FIMENQUANTO
```

FIM

## Exercício 2

a)

ED:

num, algarismo, produto INTEIRO

INICIO

produto <- 1

REPETE

LER (num)

ENQUANTO (num <= 0)

ENQUANTO (num <> 0)

algarismo <- num MOD 10

SE (algarismo MOD 2 = 1) ENTÃO

produto <- produto \* algarismo

FIMSE

num <- num DIV 10

FIMENQUANTO

ESCREVER ("O produto dos algarismos ímpares do número é: ", produto)

FIM

b)

ED:

num, algarismo, produto, n, i INTEIRO

INICIO

REPETE

LER (n)

ENQUANTO (n <= 0)

PARA i<-1 ATE n

produto <- 1

REPETE

LER (num)

ENQUANTO (num <= 0)

ENQUANTO (num <> 0)

algarismo <- num MOD 10

SE (algarismo MOD 2 = 1) ENTÃO

produto <- produto \* algarismo

FIMSE

num <- num DIV 10

FIMENQUANTO

ESCREVER ("O produto dos algarismos ímpares do número é: ",  
produto)

FIMPARA

FIM

### Exercício 3

a)

Elabore um algoritmo que inverta e apresente um número inteiro introduzido pelo utilizador.

b)

ED: num, num1, dig INTEIRO

INÍCIO

REPETIR

LER(num)

ENQUANTO (num MOD 2 <> 0 OU num MOD 3 = 0)

num1 ← 0

ENQUANTO(num <> 0)

dig ← num MOD 10

num1 ← num1 \* 10 + dig

num ← num DIV 10

FIMENQUANTO

ESCREVER("Resultado: ", num1)

FIM

### Exercício 4

ED:

num ,dig, numinv, aux INTEIRO

INÍCIO

REPETE

LER(num)

ENQUANTO num <= 0

aux <- num

numinv <- 0

ENQUANTO num <> 0

dig <- num Mod 10

numinv <- numinv \* 10 + dig

num <- num \ 10

FIMENQUANTO

SE aux = numinv ENTÃO

ESCREVER ("O número ",aux," é uma capicua")

SENÃO

ESCREVER ("O número ",aux," não é uma capicua")

FIMSE

FIM

## Exercício 5

ED:

```
num, i INTEIRO
flag BOOLEANO
```

INICIO

```
flag <- VERDADEIRO
REPETE
    LER(num)
    ENQUANTO num <= 0
    PARA i <- 2 ATE num DIV 2
        SE num MOD i = 0 ENTÃO
            flag <- FALSO
        FIMSE
    FIMPARA
    SE flag = VERDADEIRO E num <> 1 ENTÃO
        ESCRIVER (num, " é um número primo")
    SENÃO
        ESCRIVER (num, " não é um número primo")
    FIMSE
```

FIM

## Exercícios para trabalho autónomo

### Exercício 1

ED:

```
num, i INTEIRO
```

INICIO

```
LER(num)
ENQUANTO (num MOD 2 = 0 E num >= 1000 e num <= 9999)
    PARA i<-0 ATE num
        SE (i MOD 7 = 0) ENTAO
            ESCRIVER(i)
        FIMSE
    FIMPARA
    LER(num)
FIMENQUANTO
```

FIM

## Exercício 2

ED:

i INTEIRO  
dep, taxa REAL

INICIO

REPETE

LER(dep)

ENQUANTO dep <= 0

PARA i<-1 ATE 12

REPETE

LER(taxa)

ENQUANTO taxa <= 0

dep <- dep + dep\*(taxa/100)

FIMPARA

ESCREVER("Ao final do ano o valor é ", dep)

FIM

### Exercício 3

ED:

```
alt1,alt2,cent1,cent2,aux REAL
cont INTEIRO
```

INICIO

```
    REPETE
        LER(alt1,alt2,cent1,cent2)
    ENQUANTO (alt1 <= 0 OU alt2 <= 0 OU cent1 <= 0 OU cent2 <= 0)
    SE (alt1 = alt2) ENTÃO
        ESCREVER("As crianças têm a mesma altura")
    SENÃO
        SE (alt1 > alt2) ENTÃO
            aux <- alt1
            alt1 <- alt2
            alt2 <- aux
            aux <- cent1
            cent1 <- cent2
            cent2 <- aux
        FIMSE

        SE (cent1 > cent2) ENTÃO
            REPETE
                alt1 <- alt1 + cent1
                alt2 <- alt2 + cent2
                cont <- cont + 1
                ESCREVER("No final do ", cont, "º ano")
                ESCREVER("A criança mais baixa mede ", alt1")
                ESCREVER("A criança mais alta mede ", alt2)
            ENQUANTO (alt1 < alt2)
            ESCREVER("Foram precisos ", cont, " anos para a criança
mais baixa alcançar a mais alta")
        SENÃO
            ESCREVER("A criança mais baixa não pode alcançar a mais
alta")
        FIMSE
    FIMSE
FIM
```

#### Exercício 4

ED:

```
num, compMax, dig, digAnt, cont INTEIRO
primeiro BOOLEANO
```

INICIO

```
cont <- 1
compMax <- 0
primeiro <- VERDADEIRO
REPETE
    LER(num)
ENQUANTO num <= 0

    ENQUANTO num <> 0
        dig <- num MOD 10
        SE primeiro = FALSO ENTÃO
            SE dig < digAnt ENTÃO
                cont <- cont + 1
                SE cont > compMax ENTÃO
                    compMax <- cont
            FIMSE
        SENÃO
            cont <- 1
        FIMSE
    SENÃO
        primeiro <- FALSO
    FIMSE
    digAnt <- dig
    num <- num DIV 10
```

FIMENQUANTO

```
    ESCRIVER("O comprimento máximo de um segmento crescente de algarismos
é ", compMax)
")
FIM
```

#### Exercício 5

ED:

```
num, i INTEIRO
linha TEXTO
```

INICIO

```
REPETE
    LER(num)
ENQUANTO (num < 2 OU num > 20)
    PARA i <- 1 ATE num
        linha <- ""
        PARA j <- 1 ATE i
            linha <- linha + j + " "
        FIMPARA
        ESCRIVER(linha)
    FIMPARA
    PARA i <- num-1 ATE 1 PASSO -1
        linha <- ""
        PARA j <- 1 ATE i
            linha <- linha + j + " "
        FIMPARA
        ESCRIVER(linha)
    FIMPARA
```

FIM