

EXERCÍCIO 1: Faça uma função para contar a quantidade de caracteres existem em uma cadeia de caracteres, sabendo que o caractere que termina a cadeia é representado como sendo ‘\0’.

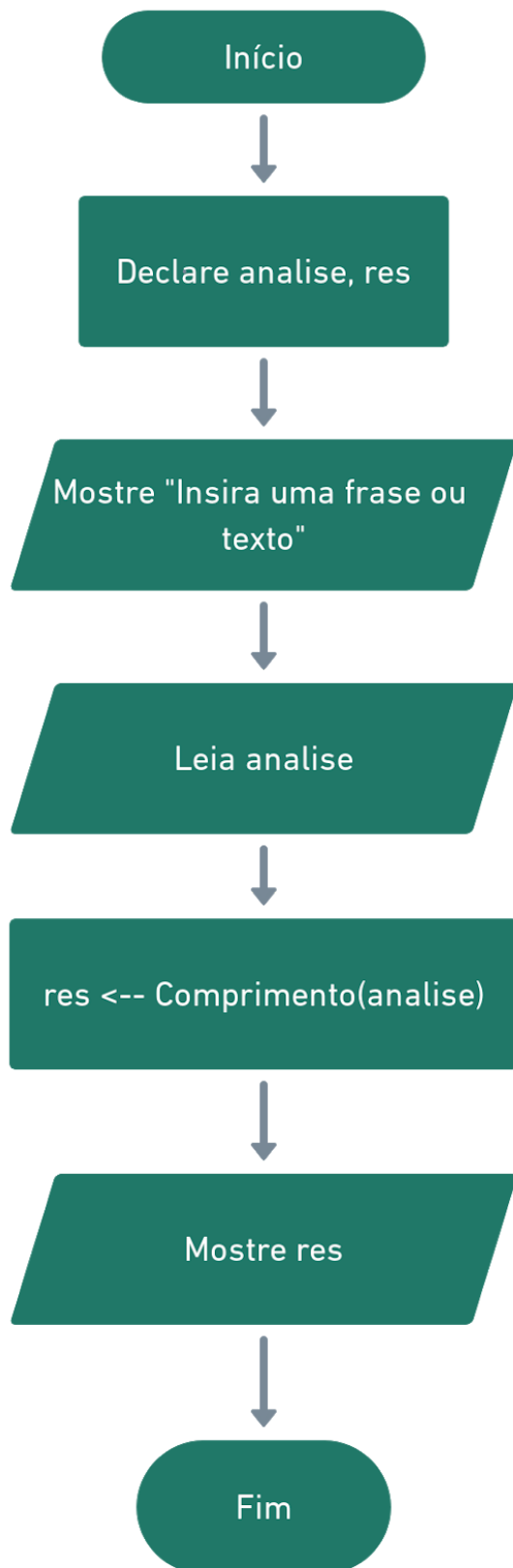
Linguagem Natural

1. Início
2. Solicitar ao usuário uma quantidade de caracteres
3. Contar a quantidade de caracteres que o usuário registrou
4. Retornar ao usuário a quantidade de caracteres.
5. Fim

Linguagem estrutural

1. Início
2. Declare analise tipo cadeia de caracteres, res
3. Mostre “Insira uma palavra, frase ou texto”
4. Leia analise
5. $res \leftarrow \text{Comprimento}(\text{analise})$
6. Mostre res
7. Fim do algoritmo

Fluxograma



Teste de mesa

EXERCÍCIO 2: Faça uma função para saber se uma dada cadeia de caracteres está vazia ou não. Ela deve retornar Verdadeiro se estiver vazia ou falso se contiver algo.

Linguagem Natural

1. Início
2. Solicitar ao usuário uma entrada de vários caracteres
3. Verificar se o usuário digitou ou não uma cadeia de caracteres
4. Retornar resultado ao usuário
5. fim do programa

Linguagem Estruturada

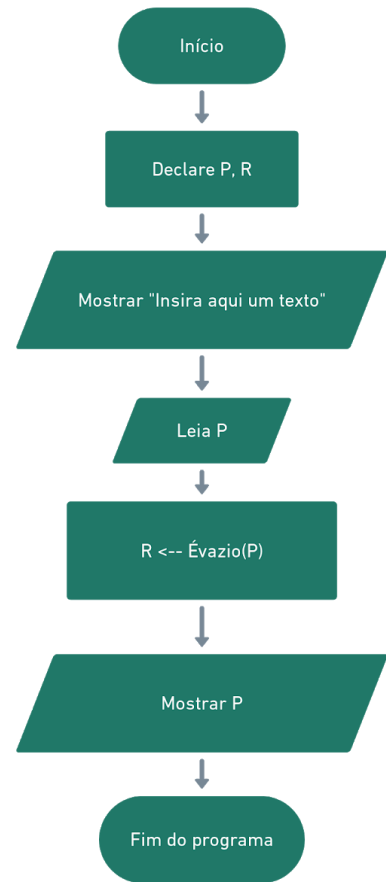
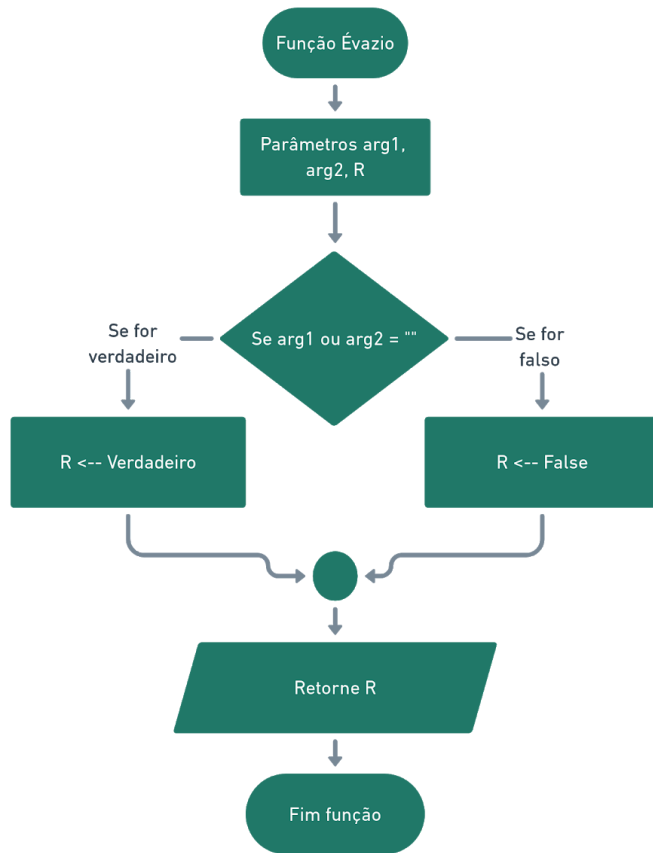
Função

1. Função Évazio
2. Parâmetro: arg1 tipo inteiro, arg2 tipo cadeia de caracteres, R
3. Se arg1 ou arg2 = ""
 - 3.1. $R \leftarrow \text{Verdadeiro}$
 - 3.2. Retorna R
4. Se não
 - 4.1. $R \leftarrow \text{False}$
 - 4.2. Retorna F
5. Fim se
6. Fim função

Programa Principal

1. Início
2. Declare P, R
3. Mostre "Insira um texto aqui"
4. Leia P
5. $R \leftarrow \text{Évazio}(P)$
6. Mostre R
7. Fim do programa

Fluxograma



Teste de mesa

Se P = “”

Se P = asd

#Passo	Comando	Variáveis	
		P	R
1	Leia P	asd	
2	R ← Évazio(P)	asd	False
...			
1	Leia P	“”	
2	R ← Évazio (P)	“”	Verdadeiro

EXERCÍCIO 3: Faça uma função para saber se uma cadeia de caracteres (s1), contendo 1 ou mais caracteres, está contida em uma outra cadeia de caracteres (s2). Deve retornar o valor -1 caso a cadeia s1 não for encontrada em s2 ou retornar a posição inicial de s1 em s2.

Linguagem Natural

1. Início
2. Pedir ao usuário que digite uma palavra ou caractere para procurar.
3. Pedir ao usuário que insira um texto qualquer para ser procurado a palavra anterior.
4. Verificar se a palavra ou caractere está contido no texto inserido pelo usuário.
5. Retornar -1 se a palavra digitada não estiver no texto. Se encontrado, retornar para o usuário a posição da palavra no texto inserido.
6. Fim do programa.

Linguagem Estruturada

1. Início
2. Declare palavra, texto, resposta
3. Mostre “Insira um texto qualquer:”
4. Leia texto
5. Mostre “Insira uma palavra ou caractere aleatório para ser procurado no texto.”
6. Leia palavra
7. resposta \leftarrow Contém(palavra, texto)
8. Mostre resposta
9. Fim do programa

Fluxograma



Teste de mesa

EXERCÍCIO 4: Faça uma função que inverta a cadeia de caracteres, ou seja, mostre a cadeia de caracteres de tras para frente. Exemplo: “faça a sua jogada.” → “adagoj aus a açaf.”.

Linguagem Natural

1. Início
2. Solicite ao usuário uma frase ou palavra para ser invertida
3. Inverta as posições até que todas os caracteres mudem de posição
4. Mostrar resultado ao usuário
5. Fim do programa

Linguagem estruturada

Função IP

1. Início - Função IP
2. Parâmetros: arg1 tipo cadeia de caractere, arg2 tipo cadeia de caractere
3. $arg2 \leftarrow “ ”$
4. Declare x, y, res
5. $x \leftarrow \text{Comprimento}(arg1)$
6. $y \leftarrow x$
7. Para $y \leftarrow y$ TO 1 <passo $\leftarrow -1$ > faça
 - 7.1. $arg2 \leftarrow arg2 + arg1[y]$
8. Fim para
9. $res \leftarrow arg2$
10. Retorne res
11. Fim Função IP

Programa principal

1. Início
2. Declare palavra tipo cadeia de caractere, resp
3. Mostre “Digite uma palavra ou frase para ser invertida”
4. Leia palavra
5. $resp \leftarrow IP(palavra)$
6. Mostre resp
7. Fim do programa principal

Fluxograma



EXERCÍCIO 5: Faça uma função para verificar se uma palavra é palíndromo ou não. Essa função vai retornar Verdadeiro se a palavra for palíndromo ou falso se não for.

Linguagem Natural

1. Início
2. Solicitar ao usuário uma palavra para verificar se é palíndroma ou não
3. Verificar se a palavra é palíndroma
4. Retornar verdadeiro ao usuário se for, retornar falso se não
5. Fim do programa

Linguagem Estruturada

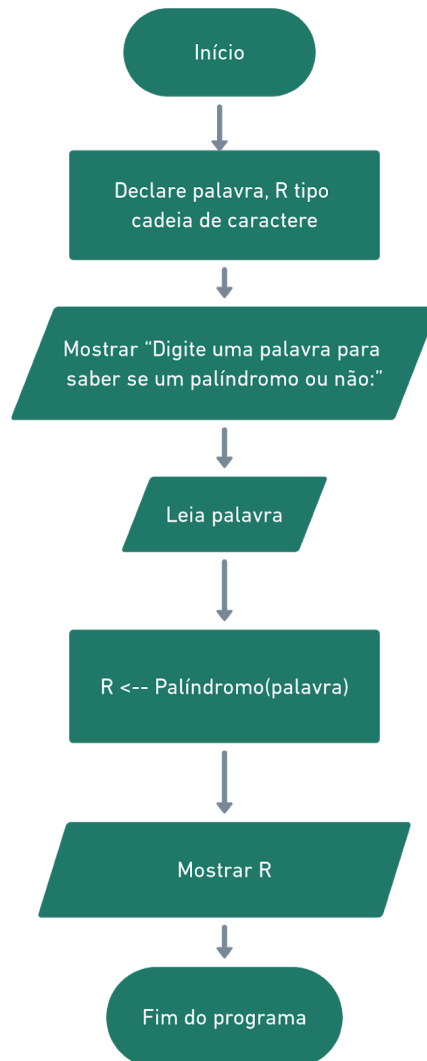
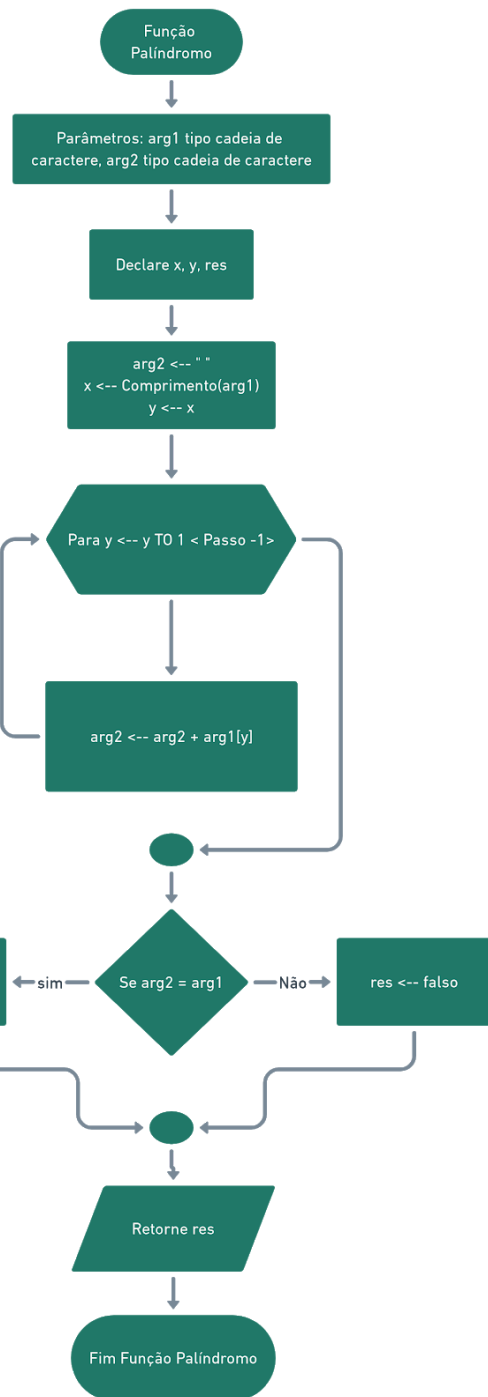
Função Palíndromo

1. Função Palíndromo
2. Parâmetros: arg1 tipo cadeia de caractere, arg2 tipo cadeia de caractere
3. $arg2 \leftarrow ""$
4. Declare x, y, res
5. $x \leftarrow \text{Comprimento}(arg1)$
6. $y \leftarrow x$
7. Para $y \leftarrow y$ TO 1 <passo $\leftarrow -1$ > faça
 - 7.1. $arg2 \leftarrow arg2 + arg1[y]$
8. Fim para
9. Se $arg2 = arg1$
 - 9.1. $res \leftarrow \text{Verdadeiro}$
10. Se não
 - 10.1. $res \leftarrow \text{falso}$
11. Retorne res
12. Fim Função Palíndromo

Programa principal

1. Início
2. Declare palavra, R tipo cadeia de caractere
3. Leia palavra (“Digite uma palavra para saber se um palíndromo ou não:”)
4. $R \leftarrow \text{Palíndromo}(palavra)$
5. Mostre R
6. Fim do programa principal

Fluxograma



EXERCÍCIO 6: Sabendo-se que o computador conta os segundos passados desde o dia 1 janeiro de 1970 e que 1972 foi um ano bissexto, faça uma função que retorne a data o formato, dia/mês/ano. O dia e mês com 2 dígitos e o ano com 4. Lembrando que o ano bissexto acontece de 4 em 4 anos, com o mês de fevereiro tendo 29 dias.

Linguagem Natural

1. Início
2. Pedir ao usuário digitar o um dia, mês e ano
3. Fazer um programa que identifique se o formato digitado pelo usuário foi válido.
4. Caso seja válido, retornar ao usuário a data no formato dd/mm/aaaa
5. Se não, retornar ao usuário “Valores inválidos, digite novamente.”
6. Fim do programa.

Linguagem estruturada

Mês 31 → Janeiro (01), Março (03) , maio (05), Julho (07), Agosto (08), Outubro (10), Dezembro (12)

Mês 30 → abril (04), Junho (06), Setembro (08), novembro (11)

Função

ano normal \leftarrow 31536000 segundos

ano bissexto \leftarrow 31622400 segundos

mês 31(são 7) \leftarrow 2678400 segundos

mês 30 (são 4) \leftarrow 2592000 segundos

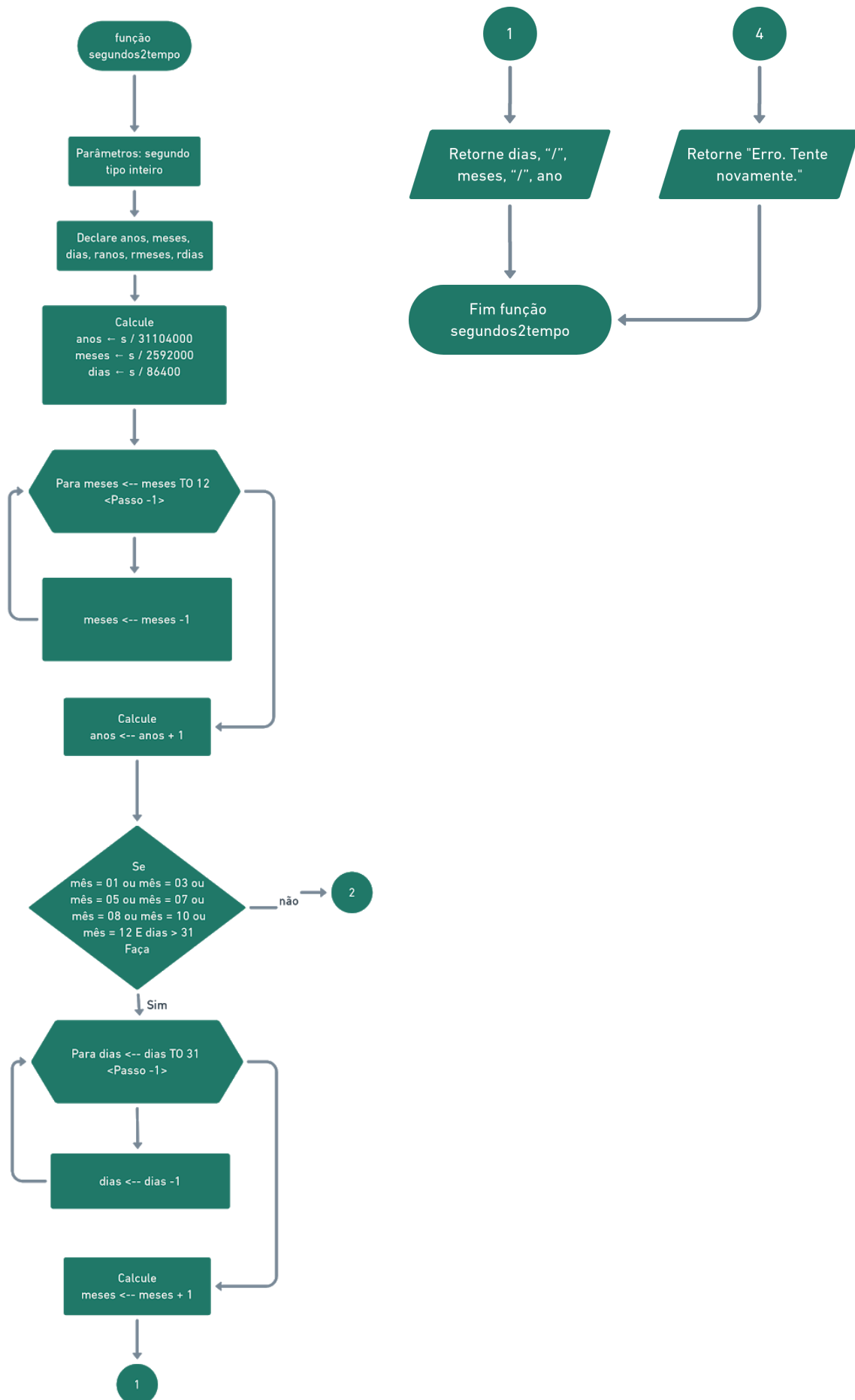
mês 28 (ou 29) \leftarrow 2419200 segundos (2505600 segundos)

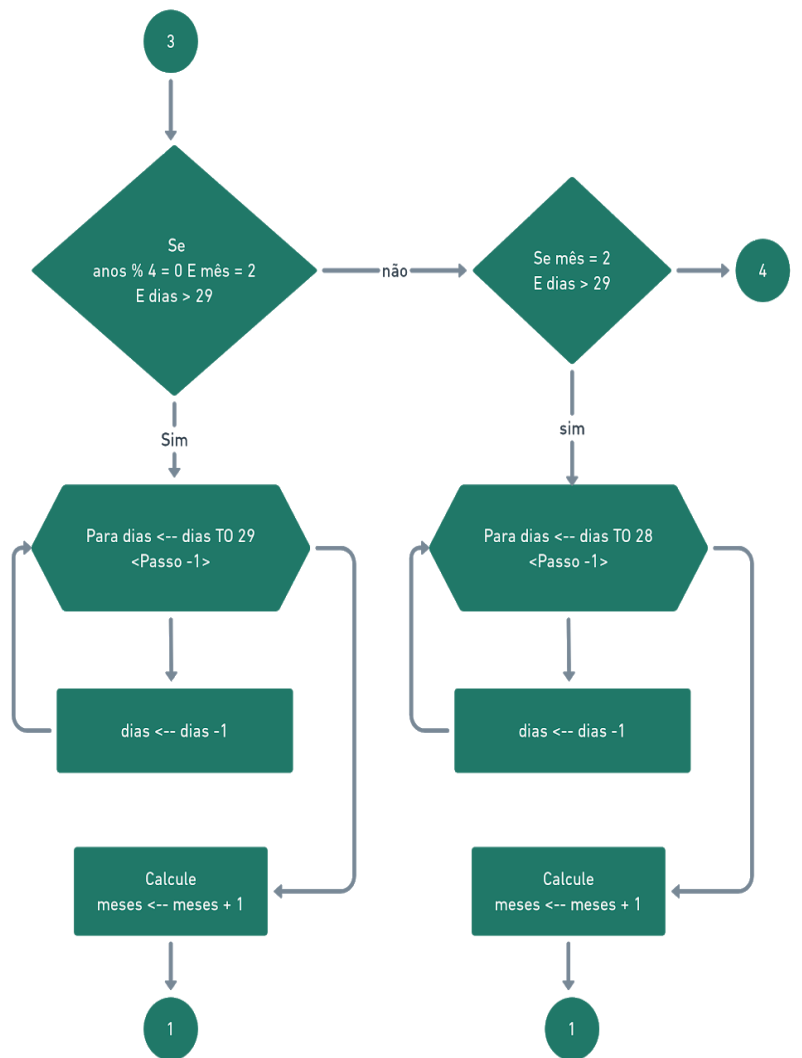
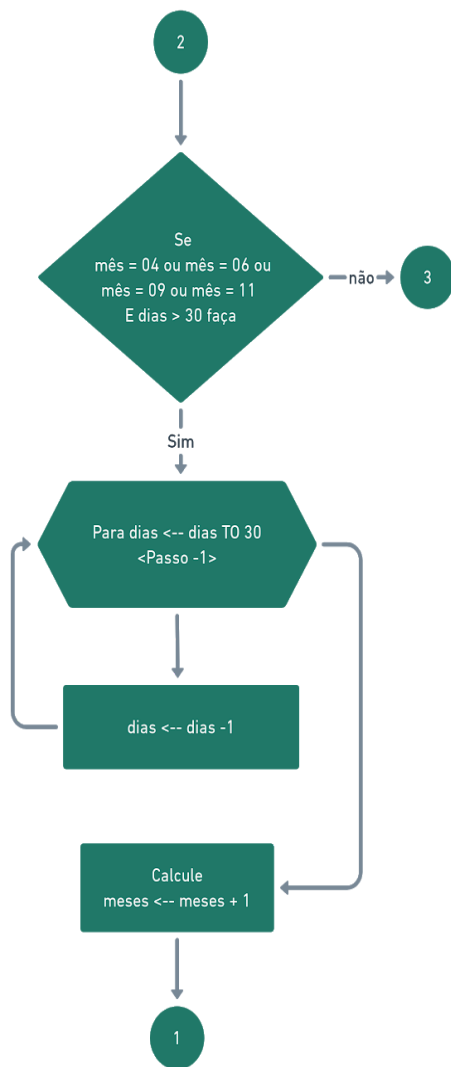
1. Função segundos2tempo
2. Parâmetros: segundos tipo inteiro
3. Declare anos, meses, dias, ranos, rmeses, rdias
4. Calcule anos $\leftarrow s / 31104000$
 - 4.1. Calcule anos $\leftarrow 1970 + \text{anos}$
5. Calcule meses $\leftarrow s / 2592000$
6. meses $\leftarrow \text{meses} + 1$
 - 6.1. Enquanto meses > 12
 - 6.1.1. Calcule meses $\leftarrow \text{meses} - 1$
 - 6.2. Fim enquanto
 - 6.3. Calcule anos $\leftarrow \text{anos} + 1$
7. Calcule dias $\leftarrow s / 86400$
8. dias $\leftarrow \text{dias} + 1$

9. Enquanto mês = 01 ou mês = 03 ou mês = 05 ou mês = 07 ou mês = 08 ou mês = 10
ou mês = 12 E dias > 31 Faça
 - 9.1. dias \leftarrow dias - 1
10. Fim enquanto
11. Enquanto mês = 04 ou mês = 06 ou mês = 09 ou mês = 11 E dias > 30 faça
 - 11.1. dias \leftarrow dias - 1
12. Fim enquanto
13. Enquanto anos % 4 = 0 E mês = 2 E dias > 29
 - 13.1.1. dias \leftarrow dias - 1
14. Retorne dias, “/”, meses, “/”, ano
15. Fim função segundos2tempo

1. Início
2. Declare s, D
3. Mostre “Digite uma quantidade de segundos para calcular a data correspondente”
4. Leia s
5. d \leftarrow segundos2tempo(s)
6. Mostre d
7. Fim programa principal

Fluxograma





EXERCÍCIO 7: Faça uma função que leia uma temperatura em graus Fahrenheit e a imprima em graus Centígrados. A conversão de graus Fahrenheit para Centígrados é obtida por $C = (5/9)(F - 32)$.

Linguagem Natural

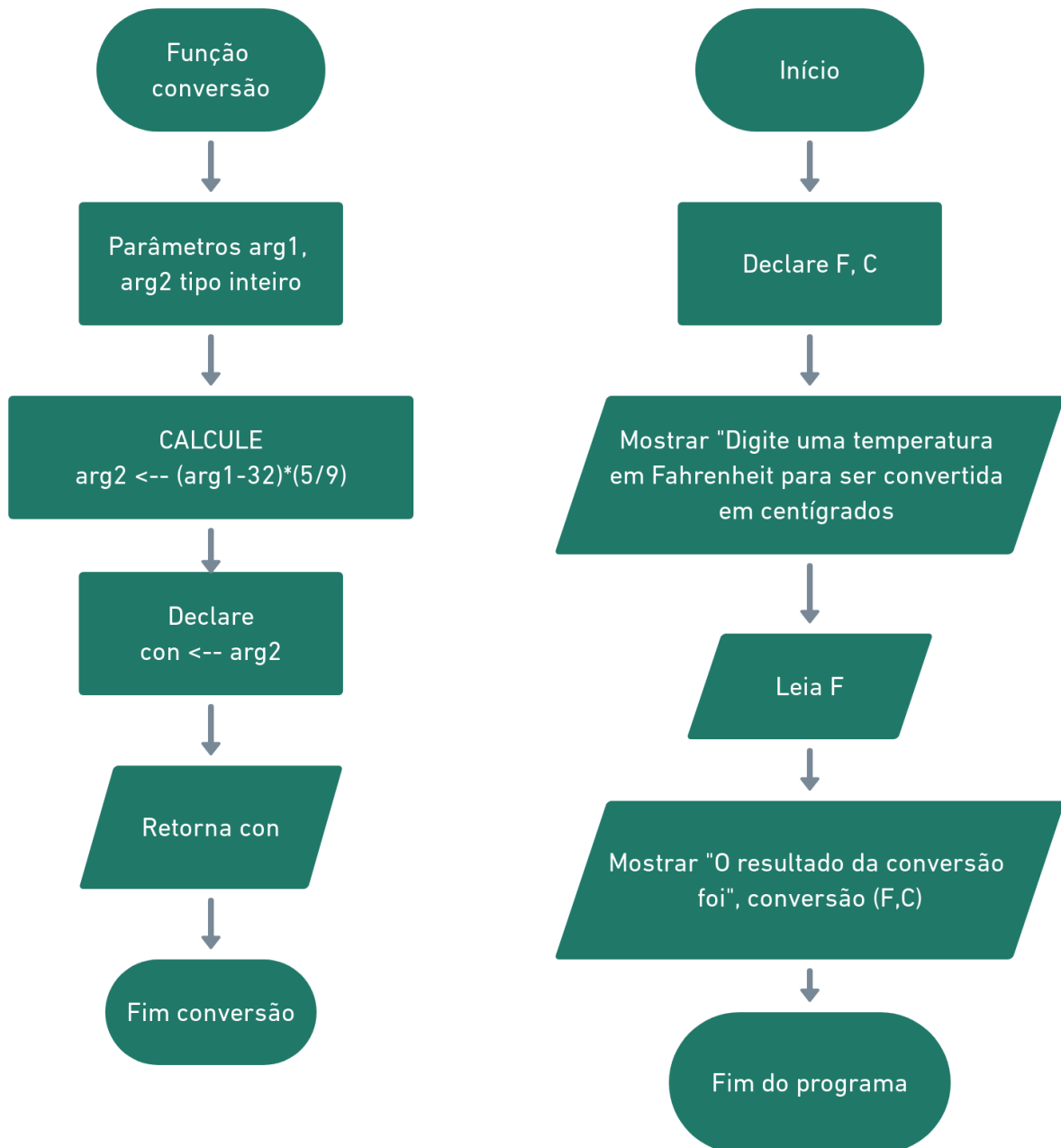
1. Início - Conversor de temperatura
2. Solicitar ao usuário uma temperatura qualquer em Fahrenheit.
3. Realizar a conversão para centígrados
4. Retornar valor ao usuário
5. Fim do programa

Linguagem Estruturada

1. Função conversão
2. Parâmetros: arg1, arg2 tipo inteiro
3. Declarar con tipo inteiro
4. Calcule $arg2 \leftarrow (arg1 - 32) (5/9)$
5. $con \leftarrow arg2$
6. Retorna con
7. Fim conversão

1. Programa Principal
2. Declare F, C tipo inteira
3. Mostrar “Digite um valor em Fahrenheit para ser convertido em Centígrados.”
4. Leia F
5. Mostrar “O resultado da conversão foi”, conversão(F,C)
6. Fim do programa principal

Fluxograma



Teste de Mesa

8. $\text{arg2} \leftarrow (\text{arg1} - 32) (5/9)$
9. $\text{con} \leftarrow \text{arg2}$
10. Se o valor digitado pelo usuário (arg1) for 100
11. Se o valor digitado pelo usuário (arg1) for 32

#Passo	Comando	Variáveis		
		arg1	arg2	con
1	$\text{arg2} \leftarrow (\text{arg1} - 32) (5/9)$	0	$(\text{arg1} - 32) * (5/9)$	-
2	$\text{con} \leftarrow \text{arg2}$	0	$(\text{arg1} - 32) * (5/9)$	arg2
3	$\text{arg2} \leftarrow (\text{arg1} - 32) (5/9)$ $\text{arg2} \leftarrow (100 - 32) * (5/9)$ $\text{arg2} \leftarrow 37,7$	100	37,7	0
4	$\text{con} \leftarrow 37,7$	100	37,7	37,7
5	$\text{arg2} \leftarrow (32 - 32) * (5/9)$	32	0	-
6	$\text{con} \leftarrow \text{arg2}$	32	0	0

EXERCÍCIO 8: Faça uma função para contar dinheiro. Os valores correspondentes aos valores de nota e moeda corrente, deverão ser passadas em dois vetores: um vetor de moedas e um vetor de notas, cada posição do vetor deverá conter a quantidade de cada valor.

Cada posição corresponderá aos seguintes valores:

Vetor moeda[6] → {1, 5, 10, 25, 50, 1}

Vetor notas[7] → {2, 5, 10, 20, 50, 100, 200}

Linguagem Natural

1. Início - conta Dinheiro
2. Solicitar ao usuário a quantidade de moedas que o mesmo tem de 1 centavo, 5 centavos, 10 centavos, 25 centavos, 50 centavos e 1 real
3. Solicitar ao usuário a quantidade de notas de 2 reais, 5 reais, 10 reais, 20 reais, 50 reais, 100 reais e 200 reais que ele tem
4. Calcular o valor total de moedas e notas
5. Retornar ao usuário o total
6. fim do programa

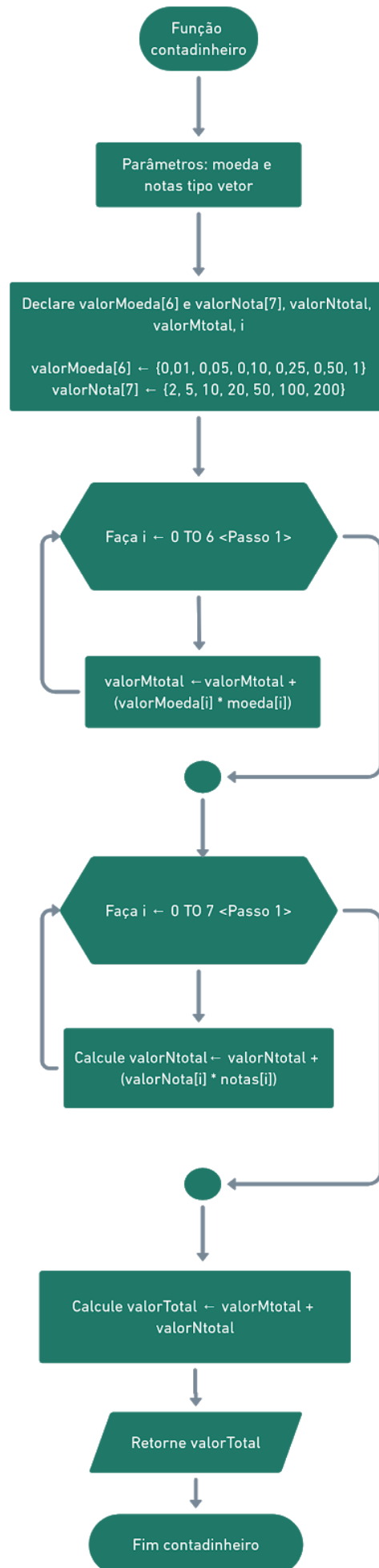
Linguagem Estruturada

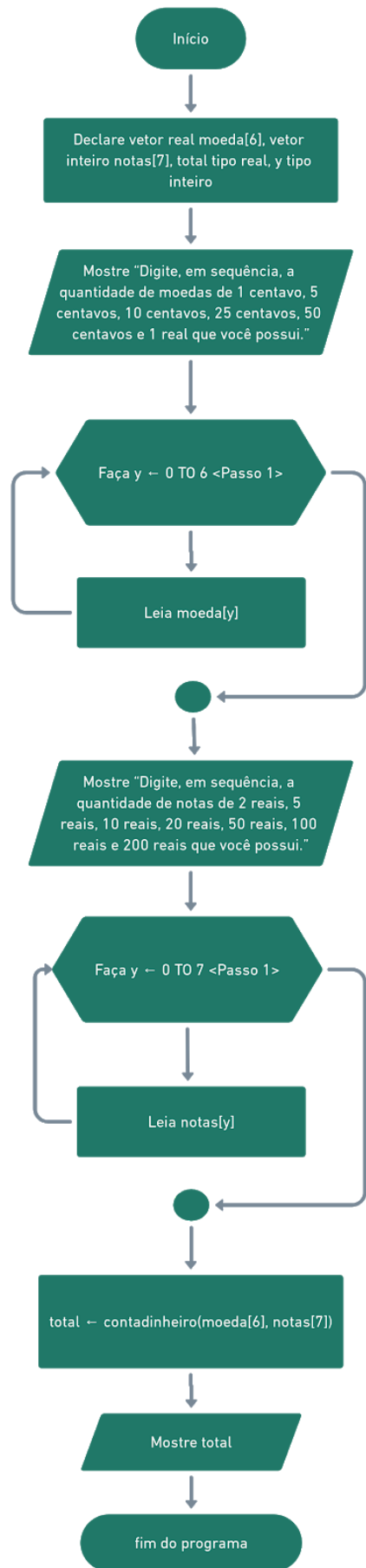
1. Função contadinheiro
2. Parâmetros: moeda e notas tipo vetor
3. Declare valorMoeda[6] e valorNota[7], valorNtotal, valorMtotal, i
4. $\text{valorMoeda}[6] \leftarrow \{0,01, 0,05, 0,10, 0,25, 0,50, 1\}$
5. $\text{valorNota}[7] \leftarrow \{2, 5, 10, 20, 50, 100, 200\}$
6. Faça $i \leftarrow 0$ TO 6 <Passo 1>
 - 6.1. $\text{valorMtotal} \leftarrow \text{valorMtotal} + (\text{valorMoeda}[i] * \text{moeda}[i])$
 - 6.2. $i \leftarrow i + 1$
7. Fim faça
8. Faça $i \leftarrow 0$ to 7 <Passo 1>
 - 8.1. Calcule $\text{valorNtotal} \leftarrow \text{valorNtotal} + (\text{valorNota}[i] * \text{notas}[i])$
 - 8.2. $i \leftarrow i + 1$
9. Fim faça
10. Calcule $\text{valorTotal} \leftarrow \text{valorMtotal} + \text{valorNtotal}$
11. Retorne valorTotal
12. Fim contadinheiro

Programa principal

1. Início
2. Declare vetor real moeda[6], vetor inteiro notas[7], total tipo real, y tipo inteiro
3. Mostre “Digite, em sequência, a quantidade de moedas de 1 centavo, 5 centavos, 10 centavos, 25 centavos, 50 centavos e 1 real que você possui.”
4. Faça $y \leftarrow 0$ to 6 <passo 1>
 - 4.1. Leia moeda[y]
 - 4.2. $y \leftarrow y + 1$
5. Fim faça
6. Mostre “Digite, em sequência, a quantidade de notas de 2 reais, 5 reais, 10 reais, 20 reais, 50 reais, 100 reais e 200 reais que você possui.”
7. Faça $y \leftarrow 0$ to 7 <passo 1>
 - 7.1. Leia notas[y]
 - 7.2. $y \leftarrow y + 1$
8. Fim faça
9. $total \leftarrow contadinheiro(moeda[6], notas[7])$
10. Mostre total
11. Fim do programa principal

Fluxograma





EXERCÍCIO 9: Faça uma função que seja informado o tempo em segundos e a função retorna o tempo em X anos, Y meses, Z dias e W minutos.

Linguagem Natural

1. Início
2. Solicitar ao usuário uma quantidade de tempo em segundos para ser convertido.
3. Converter os segundos para minutos, ou dias, ou meses, ou anos.
4. Mostrar resultado da conversão para o usuário.²
5. Fim do programa.

1. Função segundos2anos
2. Parâmetros: segundos tipo inteiro
3. Declare anos
4. Calcule $\text{anos} \leftarrow s / 31104000$
5. Retorne anos
6. Fim função segundos2anos

1. Função segundos2meses
2. Parâmetros: segundos tipo inteiro
3. Declare meses
4. Calcule $\text{meses} \leftarrow s / 2592000$
5. Retorne meses
6. Fim função segundos2meses

1. Função segundos2dias
2. Parâmetros: segundos tipo inteiro
3. Declare: dias
4. Calcule $\text{dias} \leftarrow s / 86400$
5. Retorne meses
6. Fim função segundos2dias

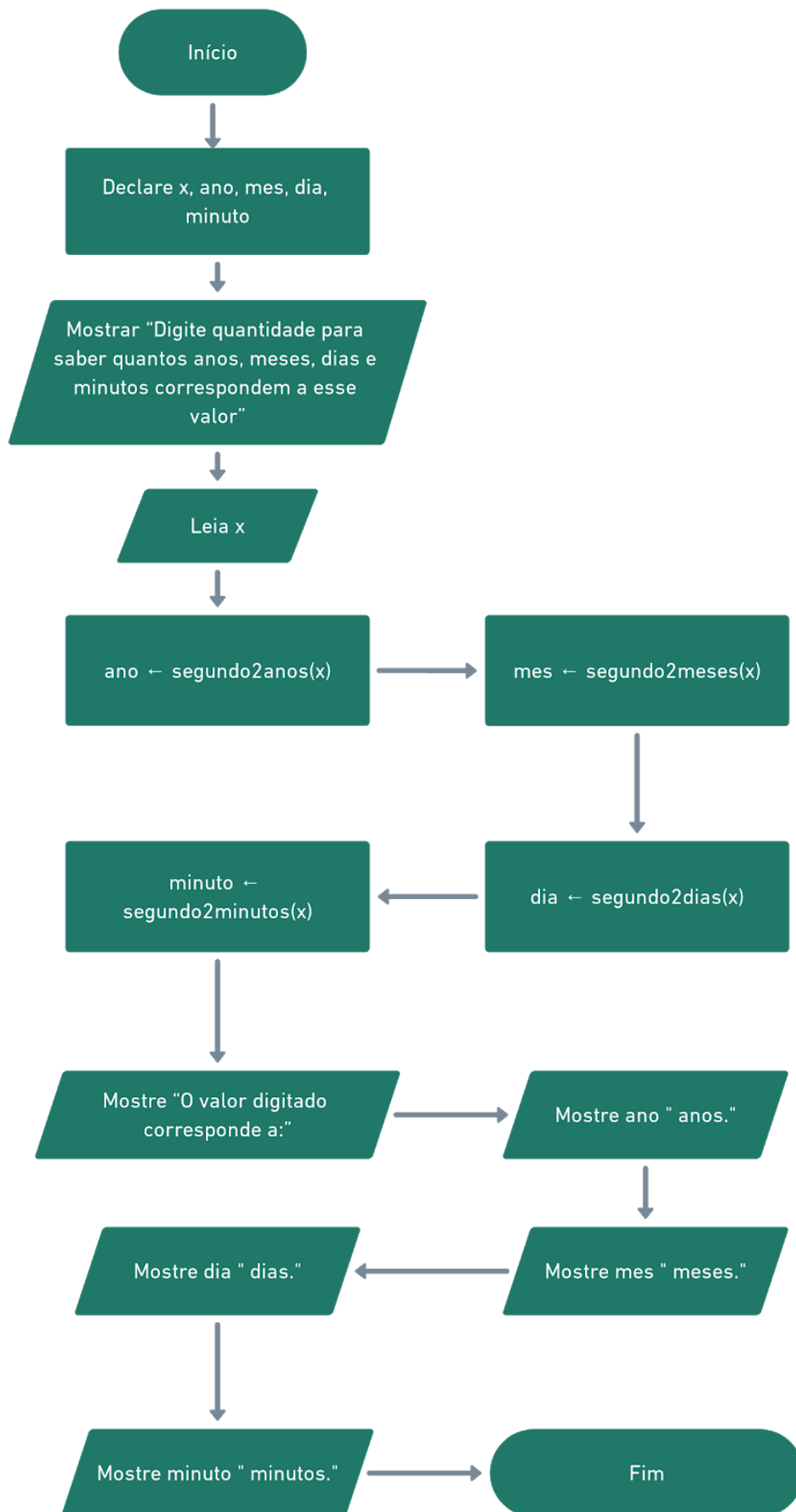
1. Função segundos2minutos
2. Parâmetros: segundos tipo inteiro
3. Declare: minutos
4. Calcule $\text{minutos} \leftarrow s / 60$
5. Retornar minutos
6. Fim função segundos2minutos

Programa principal

7. Início
8. Declare x tipo inteiro, ano, mes, dia, minuto
9. Mostre “Digite uma quantidade de segundos para saber quantos anos, meses, dias e minutos correspondem a esse valor.”
10. Leia x
11. ano \leftarrow segundo2anos(x)
12. mes \leftarrow segundo2meses(x)
13. dia \leftarrow segundo2dias(x)
14. minuto \leftarrow segundo2minutos(x)
15. Mostre “O valor digitado corresponde a:”
16. Mostre ano “anos.”
17. Mostre mes “meses.”
18. Mostre dia “dias.”
19. Mostre minuto “minutos.”
20. Fim do programa principal

Fluxograma





EXERCÍCIO 10: Faça uma função que receba duas coordenadas (x,y) e calcule a distância entre elas. Sabendo-se que a distância é calculada da seguinte forma:
 $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

Linguagem Natural

1. Início do programa - cálculo de distância entre dois pontos
2. Solicitar ao usuário as coordenadas de um ponto A qualquer (x1,y1)
3. Solicitar ao usuário as coordenadas de um ponto B qualquer (x2,y2)
4. Calcular a distância entre os dois pontos
5. Retornar ao usuário o resultado
6. Fim do programa

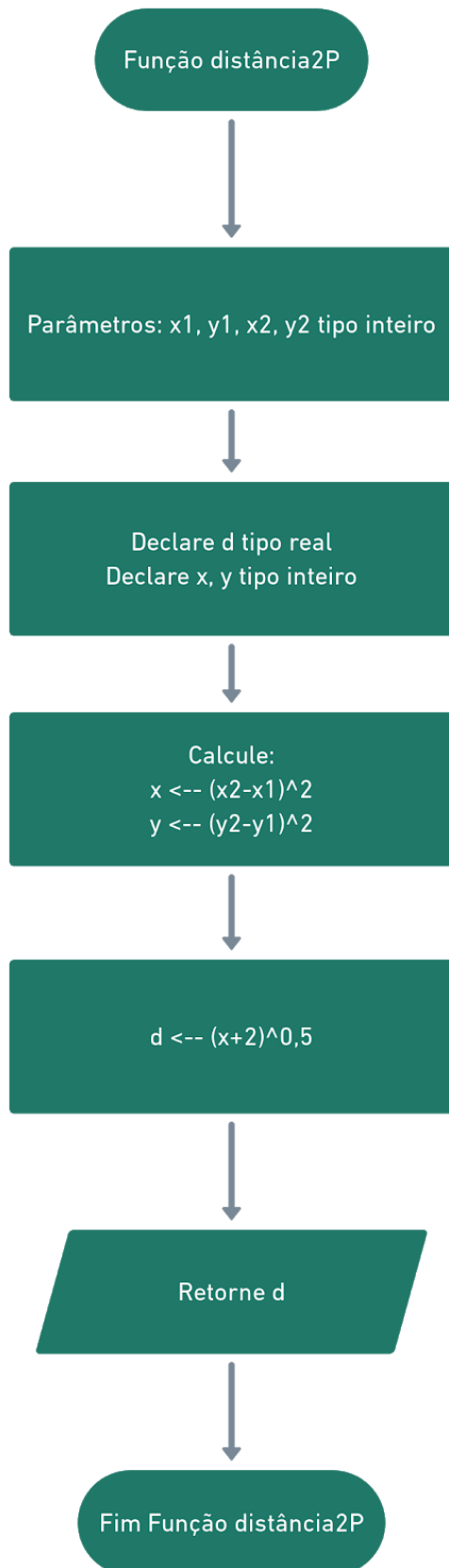
Linguagem Estruturada

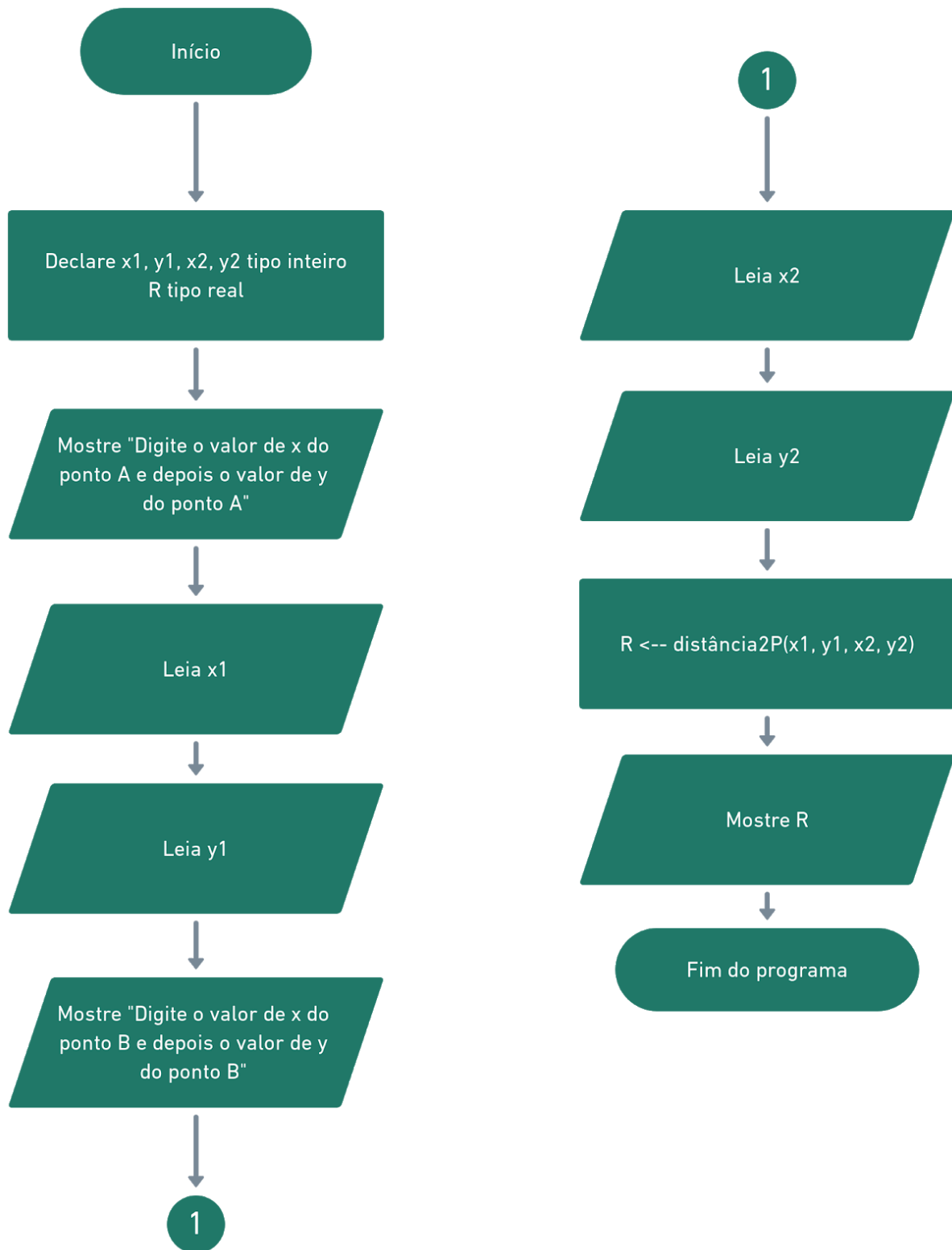
1. Função distância2p
2. Parâmetros: x1, y1, x2, y2 tipo inteiro
3. Declare d tipo real e x, y tipo inteiros
4. Calcule
 - 4.1. $x \leftarrow (x_2 - x_1)^2$
 - 4.2. $y \leftarrow (y_2 - y_1)^2$
5. $d \leftarrow (x + y)^{0,5}$
6. Retorne d
7. Fim função

Programa Principal

1. Início - Calcule a distância de AB.
2. Declare x1, x2, y1, y2 tipo inteiro, R tipo real
3. Mostre “Digite o valor de x do ponto A e depois o valor de y do ponto A”
4. Leia x1
5. Mostre “Digite o valor de y do ponto A”
6. Leia y1
7. Mostre “Digite o valor de x do ponto B”
8. Leia x2
9. Mostre “Digite o valor de y do ponto B”
10. Leia y2
11. $R \leftarrow \text{distância2P}(x_1, y_1, x_2, y_2)$
12. Mostre R
13. Fim do programa

Fluxograma





Teste de mesa