1.Criar um programa que leia o ano de nascimento de um usuário e o ano atual. Calcule e apresente a sua idade junto com mensagem correspondente:

iv. Idade < 10 anos: criança;

v. Idade >=10 e < 18: Adolescente;

vi. Idade >=18 e Idade <60: Adulto;

vii. Idade >=60 : Idoso.

Algoritmo

1. Leia o ano de nascimento (nascimento)
2. Leia o ano atual (atual)
3. Calcule a idade (idade🡨atual-nascimento)
4. Se (idade<10) então

5. Escreva (“criança”)

6. Senão

7. Se (idade >=10 e <18) então

8. Escreva (“adolescente”)

9. Senão

10. Se (idade >=18 e <60) então

11. Escreva (“adulto”)

12. Senão

13. Escreva (“idoso”)

Português estruturado

programa idade

var

nascimento, atual, idade: inteiro

inicio

leia (nascimento)

leia (atual)

idade🡨atual-nascimento

se (idade<10) então

escreva (“criança”)

senão

se (idade >=10 e <18) então

escreva (“adolescente”)

senão

se (idade >=18 e <60)

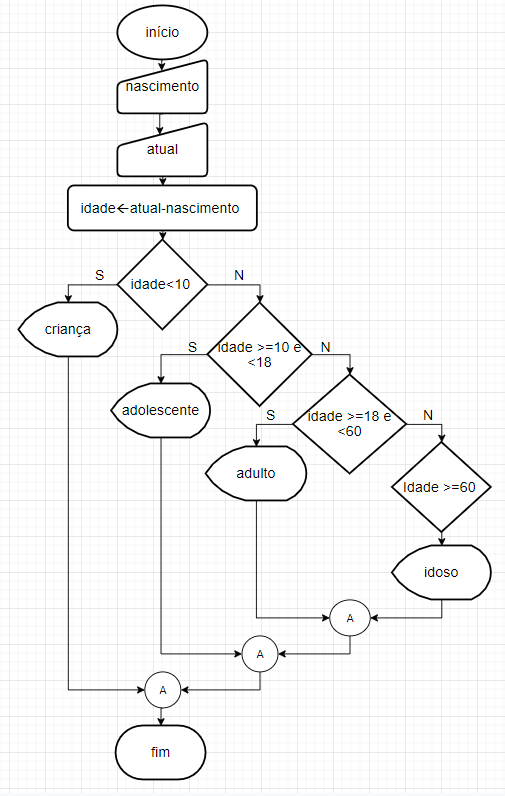
escreva (“adulto”)

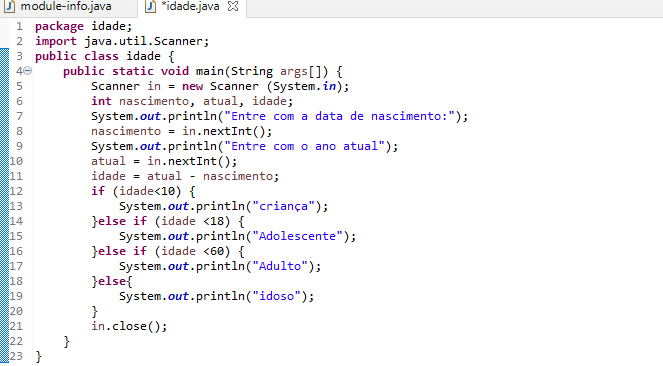
senão

escreva (“idoso”)

fim\_se

fim





2- Criar um programa que calcule e apresente o gasto médio de combustível (km por litro – km/L) de um veículo. O usuário deverá digitar a distância percorrida (em Kilômetros) e a capacidade do tanque de combustível (em litros). O gasto médio é obtido dividindo-se a distância percorrida pela capacidade do tanque. Informar se o carro é econômico ou não, segundo a regra abaixo:

a. Se o consumo for >=10 km/L: Econômico

b. Se o consumo < 10 km/L: Não econômico

Algoritmo

1. Leia a distância percorrida em quilômetros (dist)
2. Leia a capacidade do tanque de combustível em litros (cap)
3. Calcule o gasto médio (gasto🡨dist/cap)
4. Se (gasto>=10 km/l) então

5. Escreva (“econômico”)

6. Senão

7. Escreva (“não econômico”)

Português estruturado

programa gastoMedio

var

dist, cap, gasto; real

inicio

leia (dist)

leia (cap)

gasto🡨dist/cap

se (gasto>=10 km/l) então

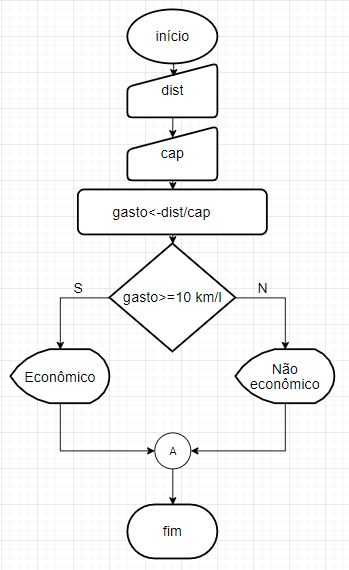
escreva (econômico)

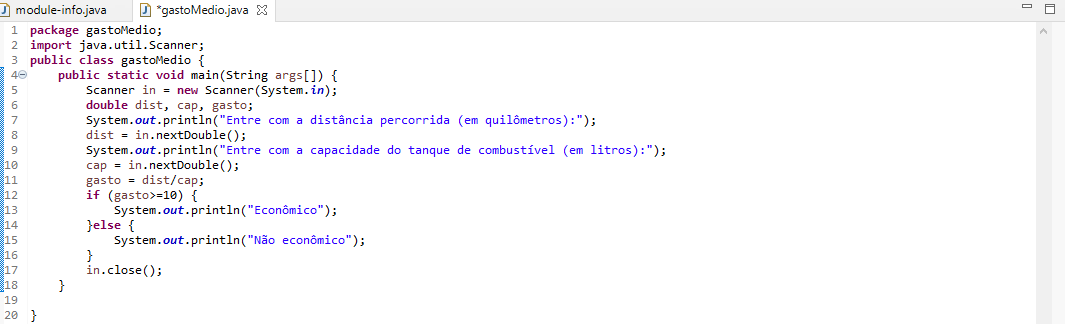
senão

escreva (não econômico)

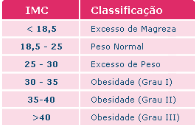
fim\_se

fim





3- Criar um programa que leia o peso e a altura de um usuário. Informar o seu IMC junto com a respectiva condição:



Algoritmo

1. Leia o peso de um usuário (peso)
2. Leia a altura de um usuário (altura)
3. Calcule o IMC (imc🡨peso/(altura\*altura))
4. Se (imc <18,5) então

5. Escreva (“excesso de magreza”)

6. Senão

7. Se (imc >=18,5 e <25) então

8. Escreva (“peso normal”)

9. Senão

10. Se (imc >=25 e <30) então

11. Escreva (“excesso de peso”)

12. Senão

13. Se (imc >=30 e <35) então

14. Escreva (“obesidade (grau I)”)

15. Senão

16. Se (imc >=35 e <40) então

17. Escreva (“obesidade (grau II)”)

18. Senão

19. Escreva (“obesidade (grau III)”)

Português estruturado

programa imc

var

peso, altura, imc; real

inicio

leia (peso)

leia (altura)

imc🡨peso/(altura\*altura)

se (imc <18,5) então

escreva (excesso de magreza)

senão

se (imc >=18,5 e <25) então

escreva (peso normal)

senão

se (imc >=25 e <30) então

escreva (excesso de peso)

senão

se (imc >=30 e <35) então

escreva (obesidade (grau I))

senão

se (imc >=35 e <40) então

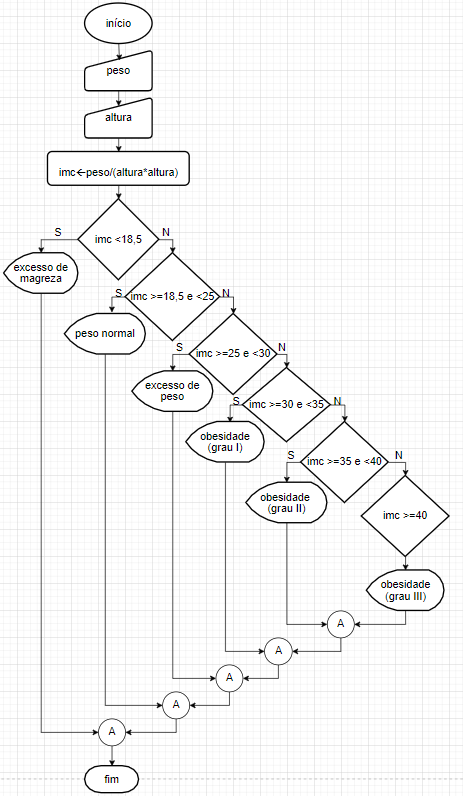
escreva (obesidade (grau II))

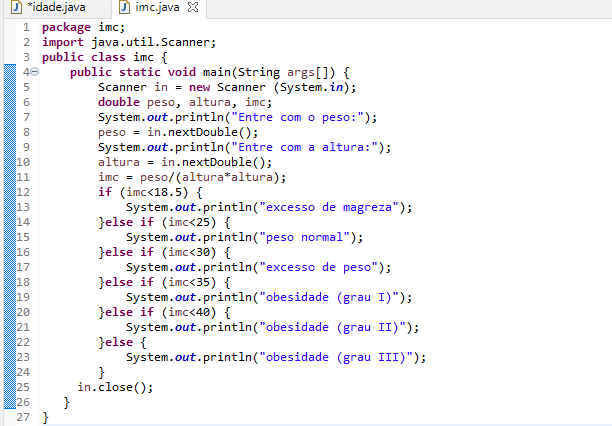
senão

escreva (obesidade (grau III))

fim\_se

fim





5. Criar um programa que leia 3 valores A, B e C para um possível triângulo e informe o tipo de triângulo lido. Verificar ainda se as medidas formam um triângulo segundo a lei:

- Para que se possa formar um triângulo é necessário que a medida de qualquer um dos lados seja menor que a soma das medidas dos outros dois.

Triângulo Eqüilátero: Três lados iguais;

Triângulo Escaleno: Três lados diferentes;

Triângulos Isósceles: Dois lados iguais e um lado diferente.

Algoritmo

1. Leia o valor A (a)
2. Leia o valor B (b)
3. Leia o valor C (c)
4. Se ( a > b+c) então

5. Escreva (“não é um triângulo”)

6. Senão

7. Se ( b > a+c) então

8. Escreva (“não é um triângulo”)

9. Senão

10. Se (c > a+b) então

11. Escreva (“não é um triângulo”)

12. Senão

13. Se ( a=b e b=c) então

14. Escreva (“triângulo equilátero”)

15. Senão

16. Se (a!=b e b!=c e a!=c) então

17. Escreva (“triângulo escaleno”)

18. Senão

19. Escreva (“triângulo isósceles”)

Português estruturado

programa triangulo

var

a, b, c; real

inicio

leia (a)

leia (b)

leia (c)

se ( a > b+c) então

escreva (“não é um triângulo”)

senão

se ( b > a+c) então

escreva (“não é um triângulo”)

senão

se (c > a+b) então

escreva (“não é um triângulo”)

senão

se ( a=b e b=c) então

escreva (“triângulo equilátero”)

senão

se (a!=b e b!=c e a!=c) então

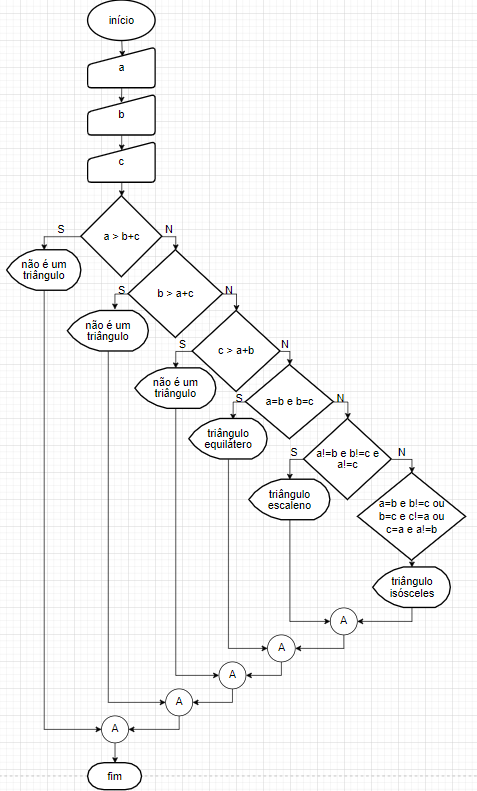
escreva (“triângulo escaleno”)

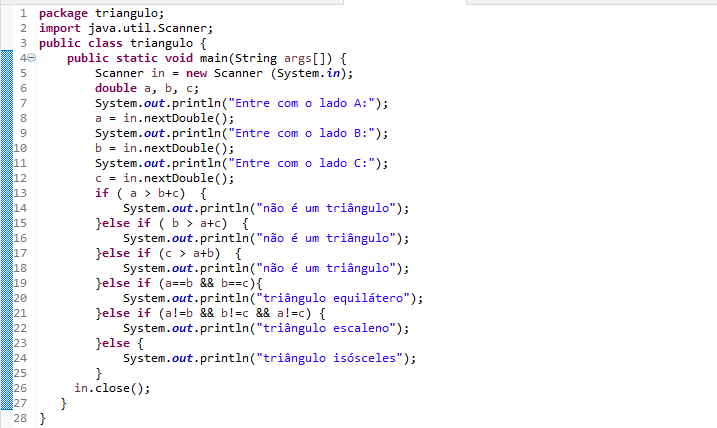
senão

escreva (“triângulo isósceles”)

fim\_se

fim





6- Criar um programa que leia três valores inteiros A, B e C e os apresente em forma crescente (menor para o maior).

Algoritmo

1. Leia o valor A (a)
2. Leia o valor B (b)
3. Leia o valor C (c)
4. Se (a <b e b <c) então

5. Escreva (“a, b, c”)

6. Senão

7. Se (a <c e c <b) então

8. Escreva (“a, c, b”)

9. Senão

10. Se (b <a e a <c) então

11. Escreva (“b, a, c”)

12. Senão

13. Se (b <c e c <a) então

14. Escreva (“b, c, a”)

15. Senão

16. Se (c <a e a <b) então

17. Escreva (“c, a, b”)

18. Senão

19. Escreva (“c, b, a”)

Português estruturado

programa valoresCrescentes

var

a, b, c; inteiro

inicio

leia (a)

leia (b)

leia (c)

se (a <b e b <c) então

escreva (“a, b, c”)

senão

se (a <c e c <b) então

escreva (“a, c, b”)

senão

se (b <a e a <c) então

escreva (“b, a, c”)

senão

se (b <c e c <a) então

escreva (“b, c, a”)

senão

se (c <a e a <b) então

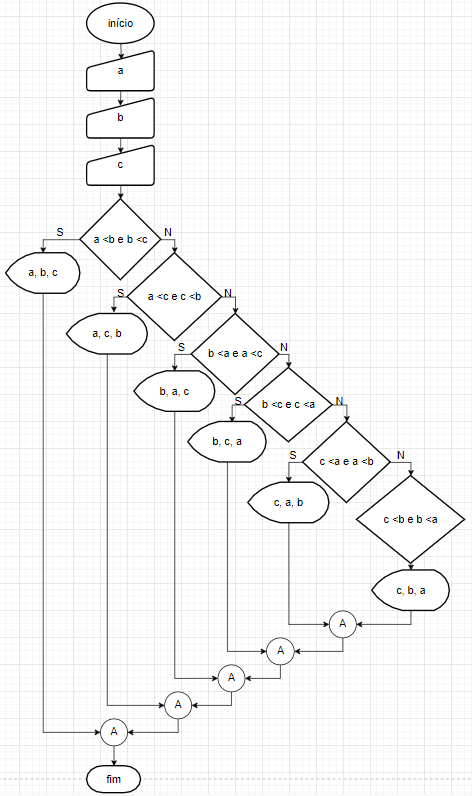
escreva (“c, a, b”)

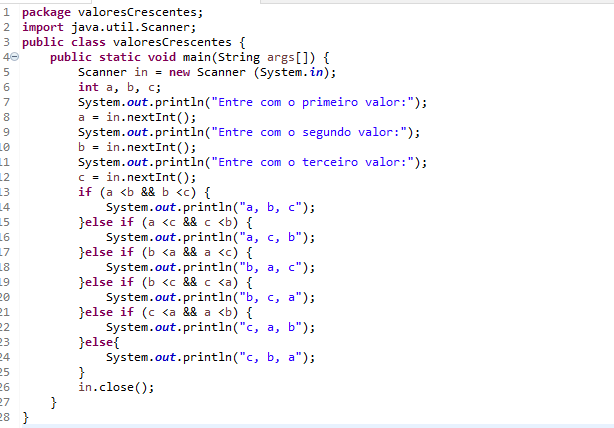
senão

escreva (“c, b, a”)

fim\_se

fim





7- Criar um programa que leia o nome e a idade de 5 usuários. Ao final apresente o nome e a idade do usuário mais velho e o nome e a idade do usuário mais novo.

Algoritmo

1. Leia o 1º nome (nome)
2. Leia a 1ª idade (idade)
3. Guardar nome lido (nomeMaior 🡨nome)
4. Guardar idade (idadeMaior 🡨idade)
5. Guardar nome lido (nomeMenor 🡨nome)
6. Guardar idade (idadeMenor 🡨idade)
7. Leia o 2º nome (nome)
8. Leia a 2ª idade (idade)
9. Se (idade >idadeMaior) então

10. Guarde na idade mais velho (idadeMaior 🡨idade)

11. Guarde no nome mais velho (nomeMaior 🡨nome)

12. Senão

13. Se (idade <idadeMenor) então

14. Guarde na idade mais novo (idadeMenor 🡨idade)

15. Guarde no nome mais novo (nomeMenor 🡨nome)

16. Leia o 3º nome (nome)

17. Leia a 3ª idade (idade)

18. Se (idade >idadeMaior) então

19. Guarde na idade mais velho (idadeMaior 🡨idade)

20. Guarde no nome mais velho (nomeMaior 🡨nome)

21. Senão

22. Se (idade <idadeMenor) então

23. Guarde na idade mais novo (idadeMenor 🡨idade)

24. Guarde no nome mais novo (nomeMenor 🡨nome)

25. Leia o 4º nome (nome)

26. Leia a 4ª idade (idade)

27. Se (idade >idadeMaior) então

28. Guarde na idade mais velho (idadeMaior 🡨idade)

29. Guarde no nome mais velho (nomeMaior 🡨nome)

30. Senão

31. Se (idade <idadeMenor) então

32. Guarde na idade mais novo (idadeMenor 🡨idade)

33. Guarde no nome mais novo (nomeMenor 🡨nome)

34. Leia o 5º nome (nome)

35. Leia a 5ª idade (idade)

36. Se (idade >idadeMaior) então

37. Guarde na idade mais velho (idadeMaior 🡨idade)

38. Guarde no nome mais velho (nomeMaior 🡨nome)

39. Senão

40. Se (idade <idadeMenor) então

41. Guarde na idade mais novo (idadeMenor 🡨idade)

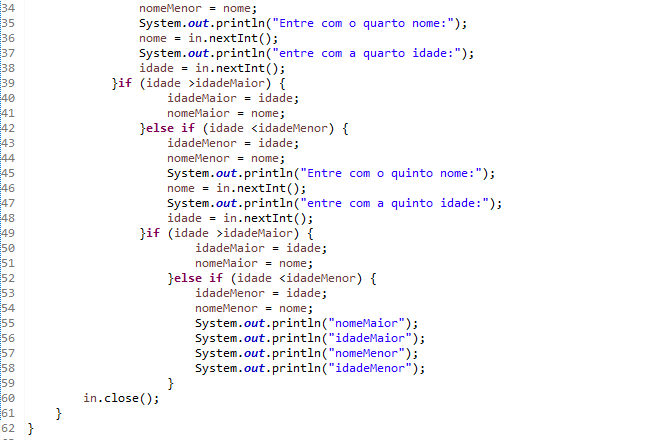
42. Guarde no nome mais novo (nomeMenor 🡨nome)

43. Apresente nome do usuário mais velho (nomeMaior)

44. Apresente idade do usuário mais velho (idadeMaior)

45. Apresente nome do usuário mais novo (nomeMenor)

46. Apresente idade do usuário mais novo (idadeMenor)



8- Criar um programa que leia duas notas para um aluno. Calcular e apresentar a sua média, sendo que:

• SE a média for menor do que 3 o aluno está REPROVADO;

• SE a média for >=6 o aluno está APROVADO;

• SE a média for >=3 e < 6 avisar via mensagem que o aluno está em EXAME. Solicitar então uma nota de EXAME, extrair uma nova média entre a média anterior e a nota de EXAME. Caso a nova média seja >=6 o aluno será APROVADO. Caso contrário está reprovado

algoritmo

1. Leia a 1ª nota (n1)
2. Leia a 2ª nota (n2)
3. Calcular a média (media🡨(n1+n2)/2)
4. Se (media <3) então

5. Escreva (“Reprovado”)

6. Senão

7. Se (media >=6) então

8. Escreva (“Aprovado”)

9. Senão

10. Leia nota de exame (ne)

11. calcule uma nova média (mediaNova🡨(media+ne)/3)

12. Se (mediaNova >=6) então

13. Escreva (“aprovado”)

14. Senão

15. Escreva (“reprovado”)

Português estruturado

programa mediaNota

var

n1, n2, media, ne, mediaNova; real

inicio

leia (n1)

leia (n2)

media🡨(n1+n2)/2

se (media <3) então

escreva (“Reprovado”)

senão

se (media >=6) então

escreva (“Aprovado”)

senão

leia (ne)

mediaNova🡨(media+ne)/3

se (mediaNova >=6) então

escreva (“aprovado”)

senão

escreva (“reprovado”)

fim\_se

fim

