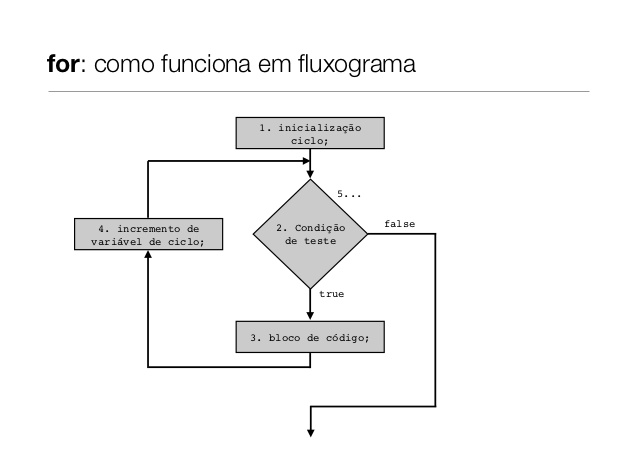
# Aula 06-Estrutura de Repetição for

## Introdução

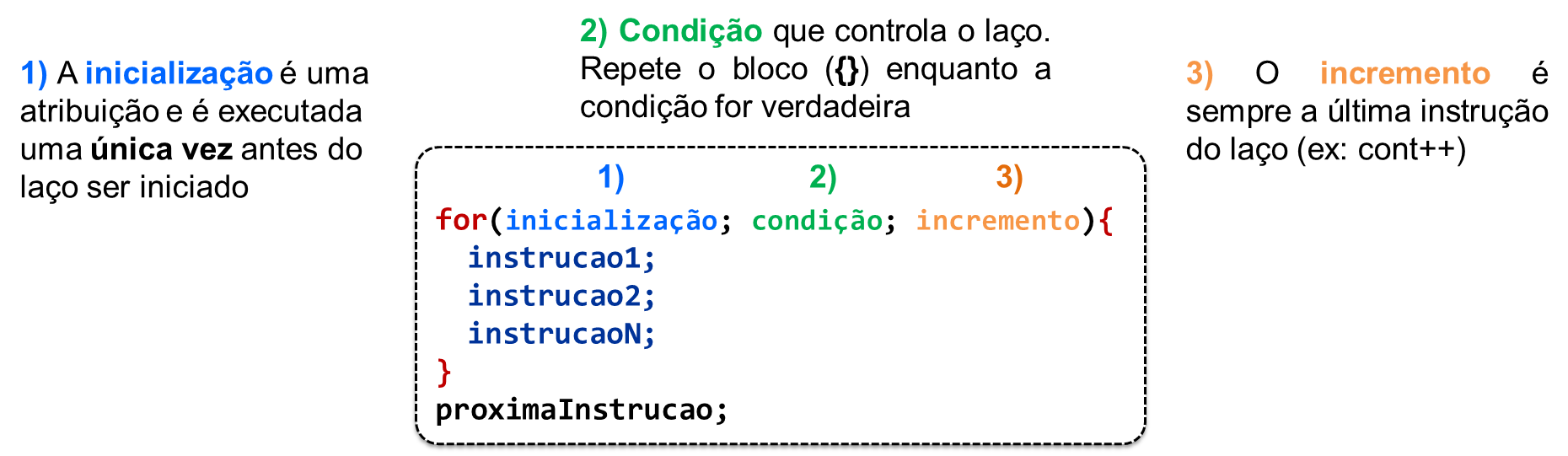
Olá, meus estudantes, sejam bem-vindos à nossa aula de Algoritmo e Programação em Java do Instituto da Oportunidade Social. Nessa aula, você vai conhecer como implementar estrutura de repetição for.

## Estruturas de repetição for

O for (que significa para) também realiza o teste lógico no início do laço. Esse comando tem uma estrutura um pouco diferente do while, mas funciona da mesma maneira. Ou seja, o programa não executará nenhuma repetição (ações programadas) sem antes testar a condição.



A estrutura de repetição for é usada quando queremos repetir um conjunto de instruções múltiplas vezes e muito útil quando sabemos o número de vezes que as instruções são repetidas. No comando for não temos um contador interno e a sua sintaxe é:

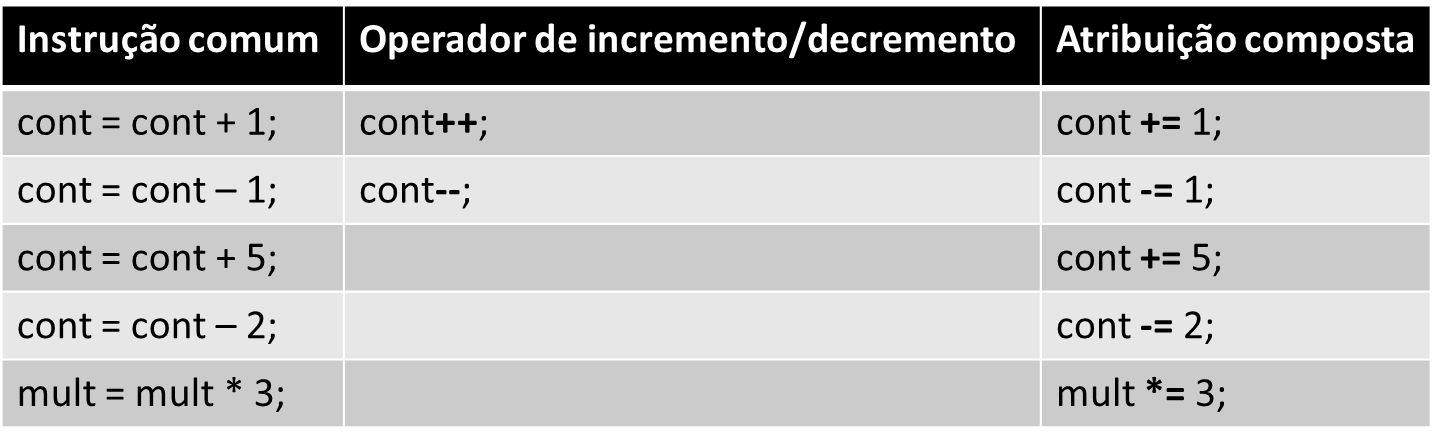


Se o laço de repetição tiver apenas uma instrução, não precisamos de usar a abertura e fechamento das chaves.



As operações de incremento e decremento podem ser realizadas:

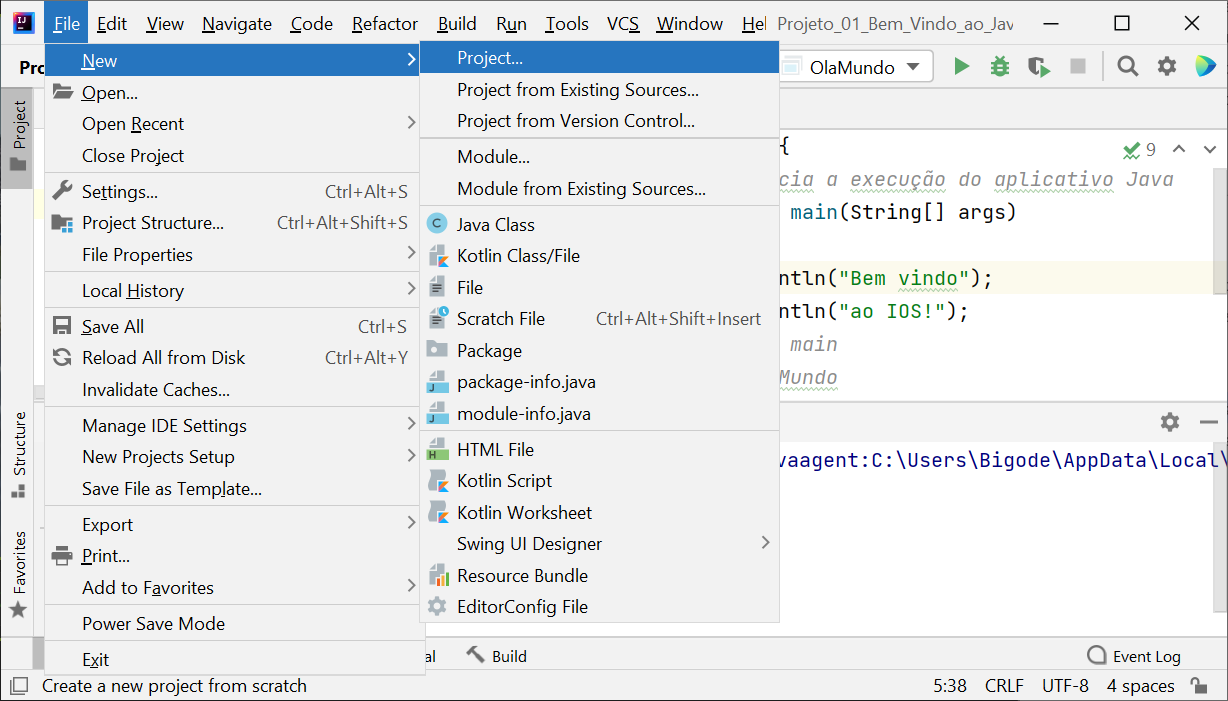
* Completas (Instrução comum)
* Com os operadores de incremento e decremento.
* Ou usando atribuição composta, contração da operação



### Vamos praticar

Vamos fazer um programa que leia a idade de 5 pessoas. Siga os passos para criar o projeto:

1. Crie um novo projeto no InterlliJ IDEA. Para isso, acesse o Menu **File**, a opção **New** e depois clique na opção **Project**.



1. As janelas seguintes são as mesmas descritas na criação de um projeto, portanto se tiver alguma dúvida consulte o material didático. Não se esqueça de dar um nome representativo para o projeto, por exemplo: Projeto\_18\_for
2. No diretório **src**, crie um arquivo para a classe. Não se esqueça de dar um nome representativo para o projeto, por exemplo: **ExemploIdade** e insira o código mostrado na Figura 1.
3. **import** java.util.Scanner;
5. **public** **class** ExemploIdade {
6. **public** **static** **void** main(String[] args) {
7. Scanner entrada = **new** Scanner(System.in);
9. **int** idade, acumuladorIdades = 0; // acumulador
10. **int** contador; // declara o contador
12. **for** (contador = 0; contador < 5; contador++) {
13. System.out.println(" Digite a sua idade ");
14. idade = entrada.nextInt();
15. acumuladorIdades += idade; // acumula as idades
16. }
18. System.out.println("A soma das idades é " + acumuladorIdades);
19. entrada.close();
20. }
21. }

Figura 1 - Código do programa utilizando for.

1. Ao executar o programa, ele irá solicitar 5 idades e imprimir a soma dessas idades no terminal.

Graphical user interface, text, application

Description automatically generated

A instrução na linha 10 da Figura 1

**for** (contador = 0; contador < 5; contador++)

esse é o nosso comando for com a inicialização (contador = 0;), condição testada (contador < 5;) e incremento (contador++).

Diagram

Description automatically generated

Sendo assim o bloco de comandos dentro do for é repetido por 5 vezes e assim solicitando o usuário entrar com 5 valores de idades.

### Vamos praticar II

Vamos fazer um programa para exibir os múltiplos de 2 menores que 1000. Siga os passos para criar o projeto:

1. Crie um novo projeto no InterlliJ IDEA. Para isso, acesse o Menu **File**, a opção **New** e depois clique na opção **Project**.

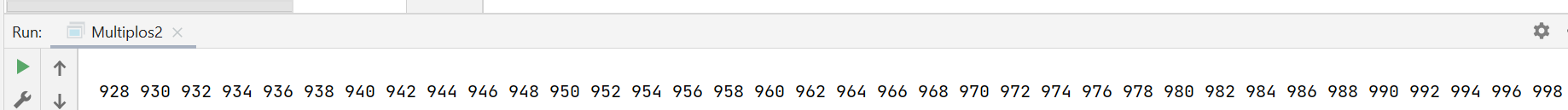
Graphical user interface, text, application, email

Description automatically generated

1. As janelas seguintes são as mesmas descritas na criação de um projeto, portanto se tiver alguma dúvida consulte o material didático. Não se esqueça de dar um nome representativo para o projeto, por exemplo: Projeto\_19\_Multiplos
2. No diretório **src**, crie um arquivo para a classe. Não se esqueça de dar um nome representativo para o projeto, por exemplo: Multiplos2 e insira o código mostrado na Figura 2.
3. **public** **class** Multiplos2 {
4. **public** **static** **void** main(String[] args) {
6. **int** numero;
8. **for** (numero = 0; numero < 1000; numero += 2) {
9. System.out.print(numero + " ");
10. }
11. }
12. }

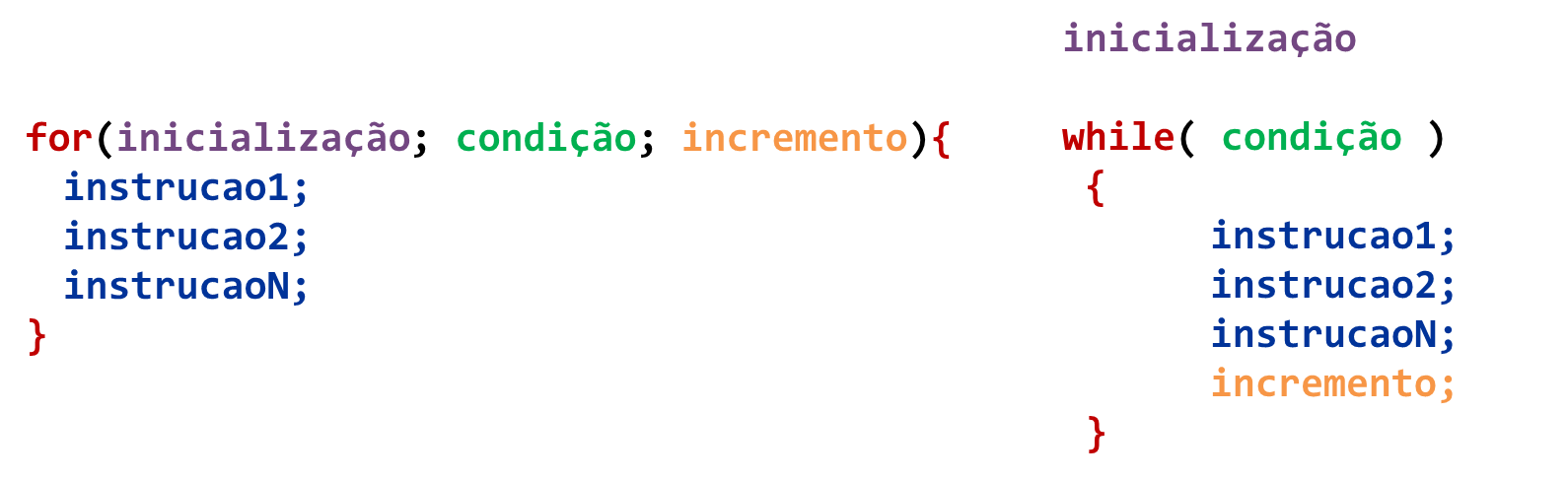
Figura 2 - Código do programa múltiplo de 2.

1. Ao executar o programa, ele irá imprimir a soma dos múltiplos de 2 menores do que 1000.



### For x while

O comando **for** e o comando **while** tem muitas semelhanças. Quando usamos esses comandos para executar as instruções um número conhecido de vezes, ambos têm a inicialização da variável de controle, uma condição testada e o incremento ou decremento da variável de controle para garantir o término do laço de repetição.



#### Vamos praticar III

Vamos fazer um programa que contenha um comando **for** e um comando **while** para executar as mesmas operações. Assim, poderemos ver como é possível usar qualquer um dos dois comandos para executar ações um número de vezes limitada. Siga os passos para criar o projeto:

1. Crie um novo projeto no InterlliJ IDEA. Para isso, acesse o Menu **File**, a opção **New** e depois clique na opção **Project**.

Graphical user interface, text, application, email

Description automatically generated

1. As janelas seguintes são as mesmas descritas na criação de um projeto, portanto se tiver alguma dúvida consulte o material didático. Não se esqueça de dar um nome representativo para o projeto, por exemplo: Projeto\_20\_for\_while
2. No diretório **src**, crie um arquivo para a classe. Não se esqueça de dar um nome representativo para o projeto, por exemplo: ForWhile e insira o código mostrado na Figura 3.
3. **public** **class** ForWhile {
4. **public** **static** **void** main(String[] args) {
6. **int** contFor, contWhile;
8. //for(inicialização; condição; incremento)
9. **for**(contFor = 0; contFor < 5; contFor++)
10. System.out.println("Valor do contFor: " + contFor);
12. contWhile = 0; // Inicialização do while
14. //while(condição)
15. **while**(contWhile < 5) {
16. System.out.println("Valor do contWhile: " + contWhile);
17. contWhile++;  // Incremento do while
18. }
19. }
20. }

Figura 3 - Código do programa ForWhile.

1. Ao executar o programa, ele irá imprimir os valores das variáveis de controle do **for** e do **while**.

Graphical user interface, text, application

Description automatically generated

A instrução na linha 7 da Figura 3

**for**(contFor = 0; contFor < 5; contFor)

mostra o comando for com a inicialização da variável de controle, a condição e o incremento da variável de controle. O mesmo pode ser visto no comando while, porém as instruções estão em diversas linhas, a inicialização na linha 10:

contWhile = 0; // Inicialização do while

a condição na linha 13:

**while**(contWhile < 5)

e o incremento na linha 15:

contWhile++;  // Incremento do while

#### Vamos praticar IV

1. Crie um novo projeto no InterlliJ IDEA. Para isso, acesse o Menu **File**, a opção **New** e depois clique na opção **Project**.

Graphical user interface, text, application, email

Description automatically generated

1. As janelas seguintes são as mesmas descritas na criação de um projeto, portanto se tiver alguma dúvida consulte o material didático. Não se esqueça de dar um nome representativo para o projeto, por exemplo: Projeto\_21\_for\_while\_2
2. No diretório **src**, crie um arquivo para a classe. Não se esqueça de dar um nome representativo para o projeto, por exemplo: ForWhile2 e insira o código mostrado na Figura 4.
3. **public** **class** ForWhile2 {
4. **public** **static** **void** main(String[] args) {
6. **int** contadorFor, contadorWhile;
8. //for(inicialização; condição; decremento)
9. **for**(contadorFor = 10; contadorFor > 0; contadorFor--)
10. System.out.print("\*");
12. System.out.println("");
13. contadorWhile = 10; // Inicialização do while
15. //while(condição)
16. **while**(contadorWhile > 0) {
17. System.out.print("\*");
18. contadorWhile--;  // Decremento do while
19. }
20. }
21. }

Figura 4 - Código do programa ForWhile2.

1. Ao executar o programa, você pode digitar quantos nomes e idades quiser até selecionar n para parar o laço de repetição.

Graphical user interface, text

Description automatically generated

A instrução na linha 7 da Figura 3

**for**(contadorFor = 10; contadorFor > 0; contadorFor--)

mostra o comando for com a inicialização da variável de controle, a condição e o decremento da variável de controle. O mesmo pode ser visto no comando while, porém as instruções estão em diversas linhas, a inicialização na linha 11:

contadorWhile = 10; // Inicialização do while

a condição na linha 14:

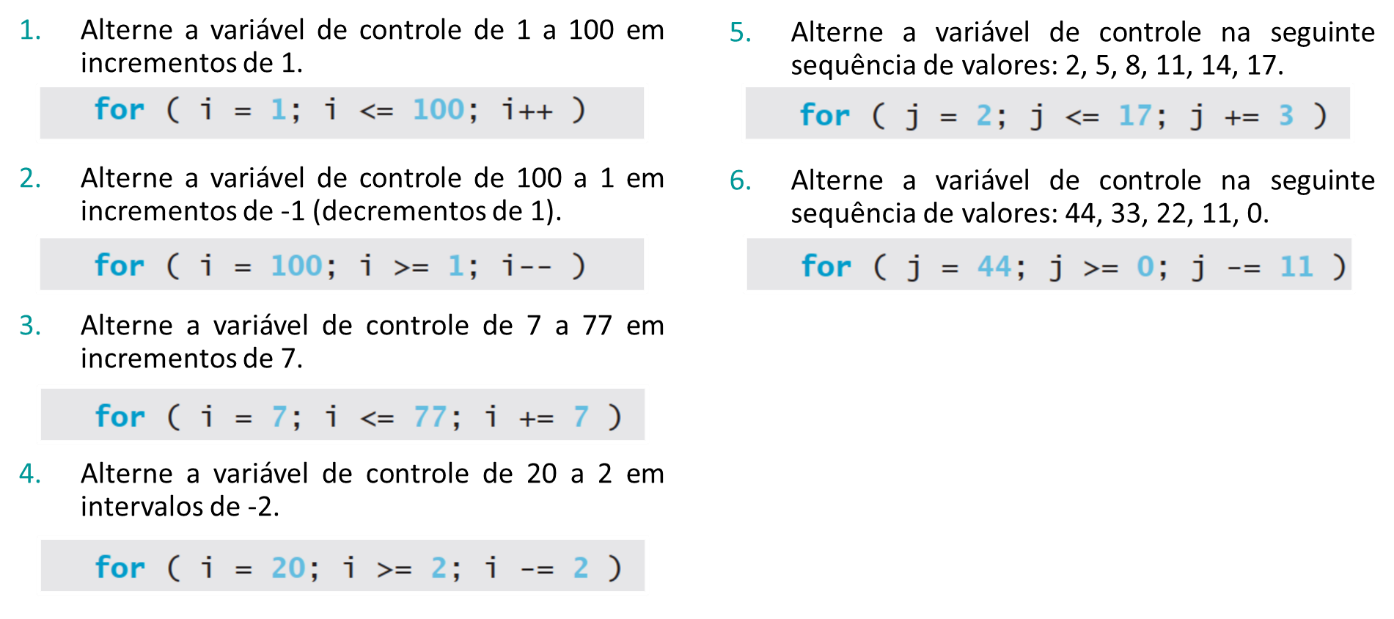
**while**(contadorWhile > 0)

e o incremento na linha 16:

contadorWhile--;  // Decremento do while

### Algumas variáveis de controle usadas no comando **for**

Os exemplos a seguir mostram métodos para a alteração da variável de controle em uma estrutura for.



## Exercícios de revisão

1. Faça um programa que imprima a seguinte sequência: 100, 99, 98, ... 1.
2. Faça um algoritmo que apresente na tela os quadrados dos números inteiros de 15 a 100.
   * Exemplo:
     + O quadrado de 15 é 225
     + O quadrado de 16 é 256
     + ...
     + O quadrado de 100 é 10000
3. Ler um número inteiro e imprimir na tela a sequência de números que vai do número 1 até o número lido:
   * Exemplo:
     + Digite um número:
     + 8
     + Sequência: 1 2 3 4 5 6 7 8..
4. Faça um algoritmo que receba a idade e a altura de 10 pessoas:
   * calcule e mostre a média das alturas daquelas com mais de 50 anos.
5. Faça um algoritmo que receba duas notas de 6 alunos, calcule e mostre:
   * A média aritmética das duas notas de cada aluno;
   * Forneça a seguinte mensagem de acordo com a nota:
     + **REPROVADO** se média é menor ou igual a 3.
     + **EXAME** se média é acima de 3 e menor que 7.
     + **APROVADO** maior ou igual a 7
   * O total de alunos aprovados.
   * O total de alunos de exame.
   * O total de alunos reprovados.
   * A média da classe.
6. Faça um algoritmo que receba a idade, a altura e o peso de 10 pessoas, calcule e mostre:
   * A quantidade de pessoas maiores de 50 anos.
   * A média das alturas das pessoas com idade entre 10 e 20 anos.
   * A porcentagem de pessoas com peso inferior a 40 quilos.