



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
AD1 de Programação Orientada a Objetos
1º semestre de 2019

Nome:

Matrícula:

Pólo:

Considere que foi contratado para implementar um gerenciador de listas (playlists) de músicas. Para tal:

- Crie uma classe que representará uma música. Para este gerenciador, precisaremos armazenar o nome da música, seu intérprete, seu compositor e a duração da música (em minutos). Declare atributos e construtor(es) de forma a facilitar a criação de objetos desta classe.
- Crie uma classe que representará uma playlist. Para tal, esta classe deverá conter um vetor para armazenar suas respectivas músicas. Qualquer detalhe para manipulação deste vetor, como um contador para controlar a quantidade de músicas alocadas, deve ser mantido interno a este classe, ou seja, sem visibilidade do mundo externo (outras classes).
- Nesta classe playlist, além do construtor, crie métodos para: inserção de uma música; obtenção da duração de uma playlist (soma das durações das músicas), e; redefina o método toString(), da classe Object, de forma que a impressão de uma playlist exiba sua duração no formato: "17h e 21min" (Lembre que músicas armazenam a duração em minutos apenas).
- Para verificar que o código funciona até o momento, crie uma playlist na main com as seguintes músicas:

Nome	Compositor	Intérprete	Duração (min)
Água de beber	Antonio Carlos Jobim	Astrud Gilberto	140
O mar serenou	Candeia	Clara Nunes	179
Rapaz Folgado	Noel Rosa	Martinho da Vila e Mart'nália	180

- Agora suponha que a playlist poderá conter vídeos também. Para armazenarmos um vídeo neste gerenciador devemos guardar: título do vídeo, assunto, nome do canal, número de visualizações e duração do vídeo.
- De forma a ilustrar a adição de propagandas ao longo do vídeo, considere a seguinte regra a ser implementada na obtenção da duração de um vídeo: i) caso o

- vídeo tenha menos de 1000 visualizações, não há tempo extra de propaganda; ii) caso o vídeo tenha entre 1000 e 10000 visualizações, há um acréscimo de 2% no tempo de propaganda; iii) acima de 10000 visualizações, o acréscimo é de 5% no tempo (pode-se trabalhar com valores inteiros, arredondados, ou com valores decimais; só observe que o tempo total de uma playlist é exibidos em horas e minutos).
- g) Como fazemos que a classe Playlist, a qual só armazenava músicas, passe agora a armazenar músicas e vídeos no mesmo vetor? Altere o programa anterior de forma que a adição de vídeos e músicas possa ocorrer no mesmo vetor.
- h) Adicione os vídeos abaixo na playlist e verifique qual o tempo de reprodução total.

Assunto	Título	Canal	Visualizações	Tempo
Musica	Despacito	Luis Fonsi	5926796	280
Musica	Gangnam Style	Officialpsy	3276192	252

Obs.: Ao longo de toda a resolução, use os conceitos de OO apresentados nas vídeo-aulas de forma a evitar redundância no código. Por exemplo, evitar a definição de atributos iguais, com a mesma funcionalidade, em classes diferentes.

GABARITO:

```
// Item g)
class Media {
    private int duracao; // Em minutos
    String titulo;

    public Media (int d, String t) {
        this.duracao = d; this.titulo = t;
    }

    public int getDuracao () {
        return this.duracao;
    }
}

// Item e)
class Video extends Media {
    String assunto;
    String titulo;
    int visualizações;
    String canal;

    public Video(int d, String a, String t, int v, String c) {
        super(d, t);
        this.assunto = a; this.visualizações = v; this.canal = c;
    }

    // Item f)
    public int getDuracao () {
        int minimo = super.getDuracao();
```

```

        if (this.visualizações < 1000)
            return minimo;
        else
            if (this.visualizações < 10000)
                return (int) (minimo * 1.02);
            else
                return (int) (minimo * 1.05);
    }
}

// Item a)
class Musica extends Media {
    String interprete;
    String compositor;

    public Musica(int duracao, String nome, String interprete, String compositor) {
        super(duracao, nome);
        this.interprete = interprete; this.compositor = compositor;
    }
}

// Item b)
class Playlist {
    Media musicas[];
    private int quantMusicas;

    public Playlist() {
        musicas = new Media[1000];
        quantMusicas = 0;
    }

    // Item c)
    public void adicionaMedia (Media m) {
        musicas[quantMusicas] = m;
        quantMusicas++;
    }

    // Item c)
    public int getDuracao() {
        int total = 0;
        for (int i=0; i<quantMusicas; i++)
            total += musicas[i].getDuracao();
        return total;
    }

    // Item c)
    public String toString() {
        int total = this.getDuracao();
        return (total / 60) + "h e " + (total - ((total / 60) * 60)) + "min";
    }
}

public class AD1_2019_1 {
    public static void main(String[] args) {
        // Item d)
        Musica aguadebeber = new Musica(140, "Agua de beber", "Astrud Gilberto",
        "Antonio Carlos Jobim");
        Musica omarserenou = new Musica(179, "O mar serenou", "Clara Nunes",
        "Candeia");
        Musica rapazfolgado = new Musica(180, "Rapaz Folgado", "Martininho da Vila
        e Mart'nália", "Noel Rosa");
    }
}

```

```

        // Item h)
        Video despacito = new Video(280, "Musica", "Despacito", 5926796, "Luis
Fonsi");
        Video gangnam = new Video(252, "Musica", "PSY - GANGNAM STYLE", 3276192,
"officialpsy");

        Playlist samba = new Playlist();
        samba.adicionaMedia(aguadebeber);
        samba.adicionaMedia(omarserenou);
        samba.adicionaMedia(rapazfolgado);
        samba.adicionaMedia(despacito);
        samba.adicionaMedia(gangnam);

        System.out.println("Duração da playlist: " + samba);
    }
}

```