

Trabalho 2 de Tópicos Avançados em Linguagens de Programação

Prof. José de Oliveira Guimarães
Veja a data de entrega no AVA

Este trabalho possui partes A, B e C. A parte seguinte envolve toda a parte anterior. Entregue apenas a parte mais abrangente que você conseguir fazer. A parte C vale 10, a B vale 8,5 e a A vale 7,5.

Parte A

Faça um protótipo com um método

```
func ageDescription: Int age -> String
```

que retorna uma string de acordo com o valor de age:

- (a) entre 0 e 3, "bebe";
- (b) entre 4 e 12, "crianca";
- (c) entre 13 e 19, "adolescente";
- (d) entre 20 e 65, "adulto";
- (e) acima de 66, "idoso".

Lance uma exceção `IdadeNegativa` se age é menor do que zero. Lance uma exceção `IdadeInvalida` se age é maior do que 127. O programa principal deve capturar estas exceções e emitir uma mensagem apropriada, sem terminar o programa.

Parte B

Modifique a parte A de tal forma que cada objeto de exceção tenha o valor da idade que causou o erro. Coloque o tratamento de exceções em um protótipo `CatchAgeErrorSoft` e use este protótipo para tratar as exceções.

Parte C

Faça um outro protótipo `CatchAgeErrorHard` para tratamento de erros de `ageDescription::`. Neste protótipo, além de imprimir as mensagens de erro, termine o programa.

Faça também um protótipo `FactoryCatch` com métodos `getCatchSoft`, `getCatchHard` e `getCatchDefault`. O primeiro método deve retornar um objeto de `CatchAgeErrorSoft`. O segundo, um objeto de `CatchAgeErrorHard`. O último, um dos dois anteriores, o que for padrão para o programa. Use um objeto retornado por `getCatchDefault` de `FactoryCatch` para tratar os erros da chamada a `ageDescription`: no método principal.