Boas Práticas de Programação

Prof. Fabrízzio Alphonsus A. M. N. Soares

fabrizzio@inf.ufg.br professor.fabrizzio@gmail.com

Instituto de Informática
Universidade Federal de Goiás

Aula 13
1 de junho de 2012

Nomes Significativos

Nomes Significativos

Há nomes por todos os lados em um software. Nomeamos diversos elementos, tais como:

- Variáveis
- Funções
- Parâmetros
- Classes
- Pacotes

Além disso, nomeamos arquivos:

- Jar
- War
- Ear

Autodocumentação

Autodocumentação

O nome de uma variável, função ou classe deve responder a todas as grandes questões. Se o nome de uma variável exige um comentário, então ele não releva seu propósito.
A variável d além de não revelar tempo, muito menos revela que este tempo é em dias.

```
1       int d; // tempo decorrido em dias
```

Use nomes que revelem seu propósito

Use nomes que revelem seu propósito

Devemos escolher nomes que revelam a mensuração e a unidade usada.
Isso pode facilitar bastante o entendimento e alteração do código.

```
1       int tempoDecorridoEmDias;
2       int diasDesdeACriacao;
3       int diasDesdeAModificacao;
4       int idadeDoArquivoEmDias;
```

Use nomes que revelam seu propósito

```
1  public List<int[]> pegueTodos() {
2      List<int[]> lista1 = new ArrayList<int[]>();
3      for (int[] x: aLista)
4        if (x[0] == 4)
5          lista1.add(x);
6
7      return lista1;
8    }
```

Listagem 1: Qual é o propósito desse código?

Use nomes que revelam seu propósito

Por que é difícil dizer o que o código faz?

Algumas questões:

- Que tipos de coisas estão em aLista?
- Qual a importância de um item na posição zero em aLista?
- Qual a importância do valor 4?
- Como eu usaria a lista retornada?

Use nomes que revelam seu propósito

Use nomes que revelam seu propósito

Digamos que esse código seja parte de um jogo que estamos desenvolvendo, chamado "Campo Minado".
Percebemos que o tabuleiro é uma lista de células chamada aLista. Por quê não chamá-la de tabuleiro?
Cada quadrado do tabuleiro é representado por um vetor simples. Mais tarde descobrimos que a posição zero armazena um vetor de status e que o valor 4 significa "marcado com uma bandeirinha".
Ao dar essas explicações como poderia ficar o código?

Use nomes que revelam seu propósito

```
1       public List<int[]> pegueCelulasMarcadas() {
2         List<int[]> celulasMarcadas = new ArrayList<int[]>();
3         for (int[] celula: tabuleiro)
4           if (celula[STATUS] == BANDEIRA)
5             celulasMarcadas.add(celula);
6         return celulasMarcadas;
7       }
```

Listagem 2: Código limpo!

Use nomes que revelam seu propósito

Use nomes que revelam seu propósito

E se a célula ao invés de ser um vetor de inteiros, fosse uma célula propriamente dita?

```
1    public List<Celula> pegueCelulasMarcadas() {
2      List<Celula> celulasMarcadas = new ArrayList<Celula>();
3      for (Celula celula: tabuleiro)
4        if (celula.estaMarcada())
5          celulasMarcadas.add(celula);
6      return celulasMarcadas;
7    }
```

Evite informações erradas!

Evite informações erradas

Variável com o nome "hp" - Talvez você queria referir-se ao valor de uma hipotenusa, mas o nome hp é o nome de uma plataforma e pode confundir outros programadores.

Variável com list em seu nome - Não nomeie uma variável como accountList a não ser que ela seja realmente um List. A palavra *list* (lista) significa muito para programadores. Se a variável não é um list, então não use isso no seu nome.

Evite informações erradas

Evite informações erradas

Evite nomes muito parecidos.

- XYZControllerForEfficient**Handler**OfStrings
- XYZControllerForEfficient**Storage**OfStrings

Evite informações erradas

Evite informações erradas

Outro típico exemplo que gera erros é o "L" minúsculo e o "O" maiúsculo. O problema é que eles se parecem com "1" (um) e "0" (zero), respectivamente.

```
1       int a = 1;
2       if ( O == 1)
3         a = O1;
4       else
5         l = 01;
```

Faça distinções significativas

Faça distinções significativas

Se você possui coisas diferentes que possuem mesmo significado, não significa que terão o mesmo nome, pois, como você já sabe, isso é proibido!
Porém, alguns programadores simplesmente transformam essas variáveis em sequências como: a1, a2, a3, etc.

```
1       public static void copyChars(char a1[], char a2[]{
2         for (int i = 0; i < a1.length; i++) {
3           a2[i] = a1[i];
4         }
5       }
```

Use nomes pronunciáveis

Use nomes pronunciáveis

O ser humano é muito bom com as palavras. Uma parte do cérebro é responsável pelo conceito de palavras. E por definição as palavras são pronunciáveis. Seria uma lástima não tirar proveito deste tipo desta importante função do nosso cérebro. Um nome que você não pode pronuncia-lo, não terá como discutir sobre tal nome sem parecer um "idiota".

```
1    int bcr3cnt = pszqint;
```

"Bem, aqui no bê cê erre três cê ene tê, temos um pê esse zê quê int".
Lembre-se: Programação é uma atividade social!

Use nomes pronunciáveis

Use nomes pronunciáveis

Veja a diferença:

```
1     class DtaRcrd102 {
2       Date genymdhms;
3       Date modymdhms;
4       String pszqint = "102";
5     }
```

Use nomes pronunciáveis

Use nomes pronunciáveis

Veja a diferença:

```
1     class Consumidor {
2       Date dataDeGeracao;
3       Date dataDeModificacao;
4       String registroId = "102";
5     }
```

Use nomes passíveis de busca

Use nomes passíveis de busca

Nomes curtos tem um problema bem particular: São difícies de serem localizados ao longo do texto. Buscar pela palavra MAX_CLASSES_PER_STUDENT é fácil, porém, experimente localizar pelo número 7. Se uma variável ou constante pode ser vista ou usada em vários lugares dentro do código, é imperativo atribuir um nome fácil para busca.

Use nomes passíveis de busca

Use nomes passíveis de busca

Compare

```
1       for (int j = 0; j < 34; j++) {
2         s += (t[j]*4)/5/
3       }
```

Use nomes passíveis de busca

Use nomes passíveis de busca

Com

```
1       int diasReaisPorDiasIdeais = 4;
2       const int DIAS_TRABALHADOS_POR_SEMANA = 5;
3       int soma = 0;
4
5       for (int j=0; j< NUMERO_DE_TAREFAS; j++) {
6         int diasTrabalhados = tarefasEstimadas[j] *
              diasReaisPorDiasIdeais;
7         int tarefasSemanais = (diasReais /
              DIAS_TRABALHADOS_POR_SEMANA);
8         soma += tarefasSemanais;
9       }
```

A notação húngara

A notação húngara

Antigamente, tinhamos que lidar com limites de tamanho para nomes. Fortran nos forçava-nos tornar a primeira letra uma indicação para o tipo.
Versões anteriores de BASIC nos permitia apenas uma letra mais um dígito. A Notação Húngara (NH) inovou essas limitações.
Atualmente, não é mais necessário fazer isto. Não é necessário definir o tipo para a variável pois eles são muito auto-explicativos.
Se quiser saber mais sobre a notação húngara acesse:
`http://pt.wikipedia.org/wiki/Nota%C\3%A7%C3%A3o_h%C3%BAngara`

Nomes de Classes e Objetos

Nomes de classes e objetos

Classes e objetos devem ter substantivos como nomes (Cliente, Pagina, Conta, Endereco).
Evite palavras como Gerenciador, Processador, Dados ou Informacao.

Nomes de Métodos/Funções

Nomes de Métodos/Funções

Os nomes de métodos/funções devem ter verbos, como postarPagamento, excluiPagina ou salvar. Deve-se nomear métodos de acesso, alteração e autenticação segundo seus valores e adicionar os prefixo get, set ou is de acordo com o padrão javabean.

1

[1] http://download.oracle.com/javase/tutorial/javabeans/

Selecione uma palavra por conceito

Selecione uma palavra por conceito

Selecione uma palavra por conceito abstrato e fique com ela.
É confuso usar as palavras pegar, obter e recuperar para o mesmo tipo de ação para classes/funções diferentes.
Você precisa ficar lembrando qual é o nome daquela função que exclui registros naquela classe ...
Qual a diferença entre *GerenciadorDeDispositivo* e *ControladorDeProtocolo*? Ambos são drivers? Fazem coisas parecidas?

Não faça trocadilhos

Não faça trocadilhos

Não use a mesma palavras para conceitos diferentes.
Exemplo:
A palavra add geralmente é usada quando se adiciona itens em uma lista. Porém, poderia significar que você está fazendo uma soma. Ou talvez, poderia significar que você está concatenando strings.
Não deixe que o contexto defina o que a palavra significa.
Adote um significado e fique com ele.

E quando não houver uma solução "à la programador"?

Se você não tem um nome técnico para dar, então, use algo que seja do domínio do problema. Assim, quem ler seu código poderá pelo menos perguntar a um especialista o significa algo no seu código.