Linguagens de Programação

Propriedades Desejáveis

Bruno Lopes

Instituto de Computação
UFF

Legibilidade: A leitura do programa é facilmente compreendida?

Redigibilidade: A implementação reflete o algoritmo? A redação é sucinta?

Confiabilidade: É fácil detectar "enganos" do programador?

Eficiência: Roda rápido?

Facilidade de Aprendizado: É enxuta?

Ortogonalidade: Conceitos podem ser combinados livremente?

Reusabilidade: É possível aproveitar partes em outros programas?

Modificabilidade: É fácil alterar programas?

Portabilidade: Roda da forma esperada em diferentes plataformas?

Instituto de Computação

Legibilidade

A leitura do programa é facilmente compreendida?

"Programs should be written for people to read, and only incidentally for machines to execute." (Structure and Interpretation of Computer Programs. Harold Abelson, Gerald Jay Sussman,Julie Sussmann)

- Comentários
- Marcadores de blocos
- Desvios Incondicionais

Instituto de Computação

Legibilidade

A leitura do programa é facilmente compreendida?

"*"Programs should be written for people to read, and only incidentally for machines to execute."* (Structure and Interpretation of Computer Programs. Harold Abelson, Gerald Jay Sussman,Julie Sussmann)

- Comentários
- Marcadores de blocos
- Desvios Incondicionais

Instituto de Computação

Legibilidade

A leitura do programa é facilmente compreendida?

"*"Programs should be written for people to read, and only incidentally for machines to execute."* (Structure and Interpretation of Computer Programs. Harold Abelson, Gerald Jay Sussman,Julie Sussmann)

- Comentários
- Marcadores de blocos
- Desvios Incondicionais

Instituto de Computação

Legibilidade

Duplicação de significado

`this` (em Java)

```
*p = (*p)*q;
```

Palavras reservadas

`if (if > then) then else` (em FORTRAN)

Instituto de Computação

Legibilidade

Duplicação de significado

`this` (em Java)

```
*p = (*p)*q;
```

Palavras reservadas

`if (if > then) then else` (em FORTRAN)

Instituto de Computação

Legibilidade

```
int x = 1;
int retornaCinco() {
  x = x + 3;
  return 5;
}

int main() {
  int y;
  y = retornaCinco () + x;
  printf ("%d\n",\y:", y);
}
```

Instituto de Computação

Redigibilidade

- Limitação nos Tipos de Dados
- Ausência de Tratamento de Exceções
- Conflito Ocasional com Legibilidade

```
void f(char *q, char *p) {
  for  (;*q=*p; q++, p++);
}
```

Instituto de Computação

Redigibilidade

- Limitação nos Tipos de Dados
- Ausência de Tratamento de Exceções
- Conflito Ocasional com Legibilidade

```
void f(char *q, char *p) {
  for  (;*q=*p; q++, p++);
}
```

Instituto de Computação

Confiabilidade

Declaração de Tipos

```
boolean u = true;
      int v = 0;
      while (u && v < 9) {
         v = u + 2;
         if  (v == 6) u = false;
      }
```

Tratamento de Exceções

```
try {
        System.out.println(a[i]);
} catch (IndexOutofBoundsException) {
        System.out.println("Erro de Indexação");
}
```

Confiabilidade

Declaração de Tipos

```
boolean u = true;
     int v = 0;
     while (u && v < 9) {
        v = u + 2;
        if  (v == 6) u = false;
     }
```

Tratamento de Exceções

```
try {
       System.out.println(a[i]);
} catch (IndexOutofBoundsException) {
       System.out.println("Erro de Indexação");
}
```

Eficiência

Tempo de execução. Características são relevantes (e.g. tipagem).

Instituto de Computação

```
for(i=0; i<n; i++)
  for(j=0; j<n; j++)
    A[i][j] = 0;

for(i=0; i<n; i++)
  for(j=0; j<n; j++)
    A[j][i] = 0;

p = &A[0][0];
t = n * n;
for(i=0; i<t; i++)
  *p++ = 0;
```

Instituto de Computação

Facilidade de aprendizado

```
c = c + 1;
c+=1;
c++;
++c;
```

Instituto de Computação

Ortogonalidade

Conceitos podem ser combinados livremente? Exemplos???

Reusabilidade

Parametrização!

Modificabilidade

```
const float pi=3.14
```

Instituto de Computação

Ortogonalidade

Conceitos podem ser combinados livremente? Exemplos???

Reusabilidade

Parametrização!

Modificabilidade

```
const float pi=3.14
```

Instituto de Computação

Ortogonalidade

Conceitos podem ser combinados livremente? Exemplos???

Reusabilidade

Parametrização!

Modificabilidade

```
const float pi=3.14
```

Instituto de Computação

Portabilidade

- Rigor no Projeto.
- Pode Contrastar com Eficiência.
- Solução de Java: máquina virtual.

Instituto de Computação