Apêndice - Códigos

1 Protótipos

```
Protótipo correto: http://cpp.sh/4sw3
#include <iostream>
_{2} using namespace std;
  void inc (int &);
6 int main ()
       int a = 2;
       inc(a);
       cout << a << endl;</pre>
11 }
void inc (int & x) { x += 1; }
Protótipo incorreto: http://cpp.sh/5z2g
#include <iostream>
_{2} using namespace std;
4 void inc ();
  int main ()
       int a = 2;
       inc(a);
       cout << a << endl;</pre>
11 }
13 void inc (int & x) { x += 1; }
```

2 Parâmetros default

```
Protótipos (correto): http://cpp.sh/7pqz6
#include <iostream>
2 #include <cmath>
  using namespace std;
   int log_x (int a, int b);
   int log_x (int a, int b = 10) {
       if (b == 10) { return log10(a); }
       else if (b == 2) { return log2(a); }
       return log(a);
10
  }
11
12
  int main ()
   {
       int a = log_x(100);
15
       cout << a << endl;</pre>
16
17
       a = log_x(8, 2);
       cout << a << endl;</pre>
19
20 }
- Protótipos (incorreto): http://cpp.sh/67btr
  #include <iostream>
2 #include <cmath>
  using namespace std;
   int log_x (int a, int b = 10);
   int log_x (int a, int b = 10) {
       if (b == 10) { return log10(a); }
       else if (b == 2) { return log2(a); }
       return log(a);
  }
  int main ()
14
       int a = log_x(100);
15
       cout << a << endl;</pre>
16
       a = log_x(8, 2);
       cout << a << endl;</pre>
19
20 }
```

```
- Ordem de definição (correto): http://cpp.sh/77ixy
#include <iostream>
2 #include <cmath>
3 using namespace std;
  void log_x (int a, int & c, int b);
  void log_x (int a, int & c, int b = 10) {
       if (b == 10) { c = log10(a); }
       else if (b == 2) { c = log2(a); }
       c = log(a);
11 }
12
int main ()
14 {
       int a;
       log_x(100, a);
       cout << a << endl;</pre>
^{17}
       log_x(8, a, 2);
19
       cout << a << endl;</pre>
21 }
- Ordem de definição (incorreto): http://cpp.sh/6fx5
#include <iostream>
2 #include <cmath>
3 using namespace std;
  void log_x (int a, int b, int & c);
  void log_x (int a, int b = 10, int & c) {
       if (b == 10) { c = log10(a); }
       else if (b == 2) { c = log2(a); }
       c = log(a);
  }
11
13 int main ()
14 {
       int a;
       log_x(100, a);
16
       cout << a << endl;</pre>
17
       log_x(8, 2, a);
       cout << a << endl;</pre>
20
21 }
```

```
- Múltiplos parâmetros default (correto): http://cpp.sh/6dbl
#include <iostream>
2 #include <cmath>
3 using namespace std;
   int f_linear (int x, int a, int b);
   int f_linear (int x, int a = 3, int b = 5) {
         return a * x + b;
   }
9
int main ()
12 {
         cout << "f(x)<sub>\(\sigma\)</sub>=\(\alpha\x\(\sigma\)+\(\sigma\)b" << endl;
13
14
         cout << a_{\sqcup}=_{\sqcup}3, b_{\sqcup}=_{\sqcup}5" << endl;
         cout << "\tf(x)_{\square}=_{\square}3x_{\square}+_{\square}5,_{\square}f(10)_{\square}=_{\square}" << f_linear(10) << endl;
16
17
         cout << "a_{\sqcup}=_{\sqcup}3, _{\sqcup}b_{\sqcup}=_{\sqcup}7" << end1;
18
         cout << "\tf(x)_=\3x_+\7,\f(10)_=\" << f_linear(10, 3, 7) << endl;
19
20 }
- Múltiplos parâmetros default (incorreto): http://cpp.sh/64ncv
#include <iostream>
2 #include <cmath>
3 using namespace std;
   int f_linear (int x, int a, int b);
   int f_linear (int x, int a = 3, int b = 5) {
         return a * x + b;
   }
10
  int main ()
   {
12
         cout << "f(x)_{\perp} = ax_{\perp} + b" << endl;
13
14
         cout << "a_{\sqcup}=_{\sqcup}3,_{\sqcup}b_{\sqcup}=_{\sqcup}5" << end1;
15
         cout << \frac{1}{t}(x)_{\perp} = 3x_{\perp} + 5,_{\perp} f(10)_{\perp} = (10) << f_linear(10) << endl;
16
17
         cout << a_{\sqcup}=_{\sqcup}3, b_{\sqcup}=_{\sqcup}7" << endl;
         cout << "\tf(x)_{\square}=_{\square}3x_{\square}+_{\square}7,_{\square}f(10)_{\square}=_{\square}" << f_linear(10, 7) << endl;
19
20 }
```

3 Ponteiros

```
- Como funções independentes: http://cpp.sh/35dj
#include <iostream>
using namespace std;
4 void inc (int &);
5 void dec (int &);
7 int main ()
       int a = 2;
       cout << a << endl;</pre>
       void (*funcao)(int&) = &inc;
       funcao(a);
       cout << a << endl;</pre>
       funcao = &dec;
       funcao(a);
17
       cout << a << endl;</pre>
19 }
21 void inc (int & x) { x += 1; }
22 void dec (int & x) { x -= 1; }
Como parâmetro: http://cpp.sh/4qays
#include <iostream>
using namespace std;
4 void inc (int &);
5 void dec (int &);
  void apply (int & x, void (*f)(int&)) { f(x); }
  int main ()
       int a = 2;
       cout << a << endl;</pre>
11
       apply(a, &inc);
       cout << a << endl;</pre>
       apply(a, &dec);
       cout << a << endl;</pre>
17
18 }
20 void inc (int & x) { x += 1; }
void dec (int & x) { x -= 1; }
```

```
- Com parâmetros default (incorreto): http://cpp.sh/2mwhj
#include <iostream>
2 #include <cmath>
  using namespace std;
   int log_x (int a, int b);
   int log_x (int a, int b = 10) {
       if (b == 10) { return log10(a); }
       else if (b == 2) { return log2(a); }
       return log(a);
  }
11
12
  int main ()
13
   {
14
       int (*1)(int, int) = &log_x;
16
       int a = 1(100);
17
       cout << a << endl;</pre>
18
19
       a = 1(8, 2);
20
       cout << a << endl;</pre>
^{21}
23 }
- Com parâmetros default (correto): http://cpp.sh/46vfk
#include <iostream>
2 #include <cmath>
3 using namespace std;
  int log_x (int a, int b);
   int log_x (int a, int b = 10) {
       if (b == 10) { return log10(a); }
       else if (b == 2) { return log2(a); }
       return log(a);
10
  }
11
12
  int main ()
13
   {
14
       int (*1)(int, int) = &log_x;
15
16
       int a = 1(100, 10);
17
       cout << a << endl;</pre>
18
19
       a = 1(8, 2);
20
       cout << a << endl;</pre>
^{21}
22 }
```

4 Sobrecarga

```
- Argumentos (correto): http://cpp.sh/6xci
#include <iostream>
using namespace std;
4 int soma (int a, int b);
  int soma (int a, int b, int & c);
7 int main ()
       int a = soma(2, 3);
       cout << a << endl;</pre>
       soma(4, 5, a);
       cout << a << endl;</pre>
15 }
int soma (int a, int b) { return a + b; }
int soma (int a, int b, int & c) { c = a + b; return c; }
 - Argumentos (incorreto): http://cpp.sh/32ptb
#include <iostream>
using namespace std;
4 int soma (int a, int b);
5 int soma (int a, int c);
  int main ()
       int a = soma(2, 3);
       cout << a << endl;</pre>
       soma(4, 5);
       cout << a << endl;</pre>
14
15 }
  int soma (int a, int b) { return a + b; }
  int soma (int a, int c) { return a + c; }
 - Argumentos default (correto): http://cpp.sh/6jr7y
#include <iostream>
using namespace std;
4 int soma (int a, int b);
5 void soma (int a, int * c, int b);
7 int soma (int a, int b = 3) { return a + b; }
  void soma (int a, int * c, int b = 5) { *c = a + b; }
  int main ()
10
  {
11
       int a = soma(2);
       cout << a << endl;</pre>
       soma(2, &a);
15
       cout << a << endl;</pre>
17
18 }
```

```
- Argumentos default (incorreto): http://cpp.sh/6penp
1 #include <iostream>
using namespace std;
  int soma (int a, int b);
  void soma (int a, int b);
  int soma (int a, int b = 3) { return a + b; }
  int soma (int a, int b = 5) { return a + b; }
  int main ()
       int a = soma(2);
12
       cout << a << endl;</pre>
13
14
       soma(2);
       cout << a << endl;</pre>
16
17
18 }
 - Tipo de retorno (correto): http://cpp.sh/6c77
1 #include <iostream>
using namespace std;
4 int soma (int a, int b);
  void soma (int a, int b, int & c);
7 int main ()
8 {
       int a = soma(2, 3);
       cout << a << endl;</pre>
10
11
       soma(4, 5, a);
12
       cout << a << endl;</pre>
13
14
  }
  int soma (int a, int b) { return a + b; }
  void soma (int a, int b, int & c) { c = a + b; }
  Tipo de retorno (incorreto): http://cpp.sh/8ix2s
1 #include <iostream>
  using namespace std;
4 int soma (int a, int b, int & c);
  void soma (int a, int b, int & c);
7 int main ()
       int a = soma(2, 3);
9
       cout << a << endl;</pre>
10
11
       soma(4, 5, a);
12
       cout << a << endl;</pre>
13
15 }
16
int soma (int a, int b, int & c) { c = a + b; return c; }
void soma (int a, int b, int & c) { c = a + b; }
```