

```

soma += x;;// Adicione o valor do elemento ao
soma.
} // Este é o fim do loop.
soma de retorno;// retorna a soma.
}
soma (primos) // => 28: soma dos primeiros 5
Prima 2+3+5+7+11
função fatorial (n) {// uma função para calcular
fatoriais
deixe o produto = 1;// Comece com um produto de 1
while (n> 1) {// repita declarações em {} while
expr in () é verdadeiro
produto *= n;// atalho para produto = produto
* n;
n--;// atalho para n = n - 1
} // Fim do loop
produto de retorno;// retorna o produto
}
Fatorial (4) // => 24: 1*4*3*2
função fatorial2 (n) {// outra versão usando um
loop diferente
Deixe eu, produto = 1;// Comece com 1
para (i = 2; i <= n; i ++) // incremento automaticamente i de
2 até n
produto *= i;// Faça isso cada vez.{ } não
necessário para loops de 1 line
produto de retorno;// retorna o fatorial
}
Fatorial2 (5) // => 120: 1*2*3*4*5

```

JavaScript suporta um estilo de programação orientado a objetos, mas é significativamente diferente da programação "clássica" orientada a objetos idiomas. O capítulo 9 abrange a programação orientada a objetos em JavaScript em detalhes, com muitos exemplos. Aqui está um muito simples exemplo que demonstra como definir uma classe JavaScript para representar Pontos geométricos 2D. Objetos que são casos desta classe têm um