



PROJETO INTEGRADO

1º SEMESTRE 2016

Documentação do Projeto Físico

A documentação a ser entregue deve conter pelo menos:

Descrição sucinta sobre o problema que o programa se propõe a solucionar.

Uma explicação sobre o problema a ser implementado através do seu programa.

Descrição sucinta sobre as linguagens e ferramentas usadas do trabalho.

Uma explicação sobre as linguagens e ferramentas usadas .

Descrição sobre os dados de entrada e saída do programa.

Uma descrição simples e clara dizendo quais são os dados de entrada e como o programa irá recebê-los. Por exemplo:

"A entrada para o programa consiste de um conjunto de descrição dos edifícios. Em cada linha haverá somente uma descrição. Cada descrição é composta por três números inteiros separados por um ou mais brancos na seguinte ordem: coordenada esquerda do edifício, altura do edifício, coordenada direita do edifício."

Listagem do programa fonte.

Listagem dos testes executados.

A listagem dos testes deve conter os dados recebidos pelo programa (dados de entrada) e os resultados apresentados (dados de saída).



Orientações sobre o Programa Fonte

Procure observar os seguintes aspectos no seu programa fonte:

Comentários.

Escreva os comentários no momento que estiver escrevendo o algoritmo. Um programa mal documentado é um dos piores erros que um programador pode cometer, e o sinal de um amador (mesmo com 10 anos de experiência). O melhor momento para se escrever os comentários é aquele em que o programador tem maior intimidade com o algoritmo, ou seja, durante a sua confecção.

Os comentários devem acrescentar alguma coisa de útil, não apenas frasear o código.

Poderiam ser colocados comentários dos seguintes tipos:

- O que faz o programa ou módulo;*
- Como chamá-lo ou utilizá-lo;*
- Significado dos parâmetros, variáveis de entrada, de saída e variáveis mais importantes;*
- Arquivos utilizados;*
- Outros módulos utilizados;*
- Métodos especiais utilizados, com referências nas quais possa se encontrar mais informações;*
- Avaliação do tempo de processamento e memória requeridos;*
- Autor, data de escrita e última atualização;*
- etc.*

Identação. Utilize identação para mostrar a estrutura lógica do programa.

A identação não deve ser feita de forma caótica. Crie algumas regras básicas de identação e procure segui-las ao escrever um programa.

Passagem de parâmetros.

Procure ser consistente na ordem e no tipo de passagem de parâmetros. Por exemplo: "Suponha que três procedimentos diferentes têm que acessar os dados de uma mesma tabela. Não faz sentido se um dos procedimentos receber a tabela por valor, o outro por referência e o último acessar a tabela como variável global."

Variáveis globais.

Evite ao máximo a utilização de variáveis globais, por que elas compartilham dados entre as diversas partes do programa de uma maneira que não é explícita, o que pode levar a erros difíceis de serem achados.



Nomes de variáveis.

Escolha nomes representativos. Os nomes de constantes, tipos, variáveis, procedimentos, funções, etc. devem identificar o melhor possível o que representam. Por exemplo, $X := Y + Z$ é muito menos claro que $Preco := Custo + Lucro$.

Utilize espaços em branco para melhorar a legibilidade.

Espaços em branco são valiosos para melhorar a aparência de um programa. Por exemplo:

- *Deixar uma linha em branco entre as declarações e o corpo do programa;*
- *Deixar uma linha em branco antes e depois dos comentários;*
- *Separar grupos de comandos que executam funções lógicas distintas por uma ou mais linhas em branco;*
- *Utilizar brancos para indicar precedência de operadores, ao invés de $A+B * C$ e bem mais legível a forma $A + B * C$;*
- *etc.*

Um comando por linha é suficiente. A utilização de vários comandos por linha é prejudicial por vários motivos, dentre eles destacam-se o fato do programa tornar-se mais ilegível e ficar mais difícil de ser depurado.

Utilize parêntesis para aumentar a legibilidade e prevenir-se contra erros.