Programação Estruturada

Aula 13 - Organização de Programas

Yuri Malheiros (yuri@ci.ufpb.br)

Introdução

- Com programas maiores começamos a separá-lo em funções
- Com a criação de múltiplas funções começamos a nos preocupar mais com a organização do programa

Variáveis locais

- Uma variável declarada no corpo de uma função é uma variável local
- Na função abaixo, a variável `soma` é uma variável local

```
int somar_digitos(int n) {
  int soma = 0;

  while (n > 0) {
    soma += n % 10;
    n /= 10;
  }

  return soma;
}
```

Variáveis locais

- Variáveis locais são alocadas quando uma função é chamada e são desalocadas quando a função termina
- Uma variável local não mantém seu valor quando a função termina
- Quando a função é chamada novamente não existe garantia que o valor dela será o valor antigo

Variáveis locais

- Uma variável local tem **escopo de bloco**
- Uma variável local só é visível do seu ponto de declaração até o fim da função
- Outras funções podem usar o mesmo nome de variável

Variáveis locais estáticas

- Colocar a palavra `static` antes da declaração da variável faz com que ela tenha um local permanente de armazenamento
    - Ou seja, ela mantém seu valor durante a execução do programa

```c
void incrementar() {
    static int i = 0;
    i++;

    printf("%d\n", i);
}
```

- A cada execução o valor exibido é incrementado
- Variáveis locais estáticas também só são visíveis dentro da função

Parâmetros

- Parâmetros da função tem o mesmo comportamento das variáveis locais
  - Mantém o valor durante a execução da função
  - Escopo de bloco

Variáveis globais

- Variáveis podem ser declaradas fora de uma função
- Um valor armazenado numa variável global fica com ela até o fim do programa
- Uma variável global tem escopo de arquivo, ela pode ser acessada em todo o arquivo

```c
int i = 0;

void incrementar() {
    i++;
    printf("%d\n", i);
}
```

Vantagens e desvantagens das variáveis globais

- Variáveis globais facilitam o acesso a variáveis por múltiplas funções
  - Por mais que isso possa ser prático, normalmente não é uma boa ideia
  - Na maioria dos casos é melhor as funções se comunicarem através dos parâmetros

Vantagens e desvantagens das variáveis globais

- Se mudarmos uma variável global, por exemplo, o seu tipo, nós precisamos checar todas as funções que usam ela para saber como essa mudança as afetará
- Se um valor incorreto for atribuído a uma variável global é mais difícil descobrir onde está o erro, pois ela pode ser utilizada em múltiplas partes do programa
- Funções que usam variáveis globais são de difícil reuso

Variáveis globais

- Esse programa deveria exibir um plano 10x10 composto por asteriscos

```
int i;

void mostrar_uma_linha(void) {
  for (i=1; i<=10; i++)
    printf("*");
}

void mostrar_todas_as_linhas(void) {
  for (i=1; i<=10; i++) {
    mostrar_uma_linha();
    printf("\n");
  }
}
```

- Qual o bug?

Blocos

- Em vários comandos da linguagem C nós utilizamos: `{ comandos }`
- De forma mais geral, podermos ter:

```
{
  declarações
  comandos
}
```

- Isto é chamado de **bloco**

Blocos

```
if (i>j) {
  int temp = i;
  i = j;
  j = temp;
}
```

- Variáveis declaradas num bloco tem **escopo do bloco**
- O corpo de uma função é um bloco
- Declarar variáveis dentro de um bloco ajuda na organização, evita conflito de nomes de variáveis e deixa o programa mais fácil de entender e manter

Escopo

- Em um programa talvez um mesmo nome tenha significados diferentes

```c
int i = 1;

int main() {
    int i = 2;
    printf("%d\n", i);

    return 0;
}
```

- Qual o valor exibido?

Escopo

- As regras de escopo ajudam o programador (e o compilador) a determinar qual o significado de um identificador num determinado ponto do programa
- A regra mais importante do escopo é:
    - Quando uma declaração dentro de um bloco cria um identificador que já era visível, essa nova declaração "esconde" o valor antigo. No final do bloco o identificador ganha seu significado anterior.

Escopo

```c
int i;                          /* #1 - Escopo de arquivo */

void f(int i) {                 /* #2 - Escopo de bloco */
  i = 1;
}

void g(void) {
  int i = 2;                    /* #3 - Escopo de bloco */

  if (i > 0) {
    int i;                      /* #4 - Escopo de bloco */
    i = 3;
  }
  i = 4;
}

void h(void) {
  i = 5;
}
```

Escopo

- A variável `i` é utilizada cinco vezes
- `i = 1` se refere a `#2`
- `i > 0` se refere a `#3`
- `i = 3` se refere a `#4`
- `i = 4` se refere a `#3`
- `i = 5` se refere a `#1`

Organizando um programa em C

- Um programa pode conter:
  - Diretivas
  - Definições de tipos
  - Declarações de variáveis globais
  - Protótipos de funções
  - Definições de funções

Organizando um programa em C

- Existem várias formas de organizar um programa
- Uma possível ordem:
  - Diretivas `#include`
  - Diretivas `#define`
  - Definições de tipos
  - Declarações de variáveis globais
  - Protótipos de funções
  - Função `main`
  - Outras funções

Exemplo

- Para mostrar como um programa pode ser organizado, vamos fazer um programa um pouco mais complexo que os exemplos anteriores
- O programa vai ler e classificar uma mão no Poker
- Cada carta tem um naipe e um valor

Exemplo

- As categorias possíveis são:
  - Straight flush (um straight e um flush ao mesmo tempo)
  - Four-of-a-kind (quatro cartas do mesmo valor)
  - Full house (um three-of-a-kind e um par)
  - Flush (cinco cartas do mesmo naipe)
  - Straight (cinco cartas com valores consecutivos)
  - Three-of-a-kind (três cartas do mesmo valor)
  - Dois pares
  - Par (duas cartas do mesmo valor)
  - Carta mais alta (qualquer outra mão)

Exemplo

- Se a mão cair em mais de uma categoria, o programa escolhe a melhor
- Vamos abreviar a representação dos valores e naipes
- Valores: 2, 3, 4, 5, 6, 7, 8, 9, t, j, q, k, a
- Naipes: c, d, h, s

Exemplo

- Se o usuário entrar com uma carta inválida ou uma mesma carta duas vezes, o programa deve ignorar a carta, exibir uma mensagem de erro e pedir para o usuário entrar novamente com uma carta

Exemplo

- Nosso programa tem três tarefas principais:
  - Ler uma mão de cinco cartas
  - Analisar a mão
  - Exibir a classificação