

```

watershed.c > ...
1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4  #include "../utils/image
5
6  /*-----
7   * Fila de prioridade
8   *-----
9  typedef struct no *ptno;
10 typedef struct no
11 {
12     int i, j;
13     ptno next;
14 } no;
15
16 /*-----
17 * init *Q new in
18 * [a:]<[b:]> [c:] ->[a:]
19 * ^-^
20 *-----
21 void InsQ(ptno *Q, int i, int j)
22 {
23     ptno new = malloc(sizeof(no))
24     new->i = i;
25     new->j = j;
26     if (!(*Q))
27         new->next = new;
28 }

```

Algoritmos Estruturados

Luiz Eduardo da Silva

Algoritmos e Estrutura de Dados I

Ciência da Computação

UNIFAL-MG

Agenda

- 1 Algoritmos Estruturados
 - Tipos dos problemas
 - Objetivos das técnicas
 - Desenvolvimento Estruturado
 - Módulos

Agenda

- 1 Algoritmos Estruturados
 - Tipos dos problemas
 - Objetivos das técnicas
 - Desenvolvimento Estruturado
 - Módulos

Algoritmos Estruturados

- O tempo de processamento de computadores era muito caro (\$\$\$).
- Antes de usar o computador para resolver um problema é necessário:
 - Planejar
 - Desenvolver o algoritmo
 - Testar, ou seja, antecipar as situações que podem ocorrer quando for posto em execução

Tipos de problemas

- **Problemas Simples:** Não existe dificuldade em desenvolver o algoritmo adequado.
- **Problemas Complexos:** Necessitam de técnicas como o desenvolvimento estruturado de algoritmos.

Objetivos das técnicas

- Facilitar o **desenvolvimento** dos algoritmos;
- Facilitar o **entendimento** pelos humanos;
- Antecipar a comprovação da sua **correção**;
- Facilitar a sua **manutenção** e sua **modificação**;
- Permitir o seu entendimento por uma **equipe** de pessoas.

Desenvolvimento estruturado

- Os algoritmos devem ser desenvolvidos por refinamentos sucessivos (**top-down**);
- Os refinamentos devem ser módulos **independentes**, com pouco vínculo com os outros módulos;
- Nos módulos deve-se usar **um número limitado de comandos** e de estrutura de controles.
- Deve-se buscar **simplicidade** e **clareza**

Módulos

- Contorna a **limitação humana** para compreender a complexidade;
- Cada módulo pode ser desenvolvido de forma **independente**;
- Cada módulo pode ser desenvolvido por um elemento da equipe e depois **integrado** aos módulos dos outros membros.