SOFTWARE DE RECONHECIMENTO E CONDUÇÃO DE VEÍCULO EM PISTA DE ROLAMENTO

Tarcísio B. Prates - tarcisiobatistaprates@gmail.com
Francisco A. Gonçalves - franciscoabreu1408@gmail.com
Centro Federal de Educação Tecnológica de Minas Gerais/Campus II, Avenida Amazonas nº 7675, Nova Gameleira, 30510-000 - Belo Horizonte - Minas Gerais

1. INTRODUÇÃO

Durante o curso de Algoritmo e Estrutura de Dados I, foi proposto a criação de um software que identificasse uma pista de rolamento, seu meio e direção. Para identificação da pista, foi fornecido um *array* contendo as cores de uma imagem, que a partir delas, o software deveria identificar a pista, calculando seu meio e, a partir de uma imagem inteira, definir a direção da pista. A resolução do problema, se deu por meio da utilização dos conceitos de estruturas de dados, mais especificamente das listas, tema amplamente abordado no curso, o qual foi introduzido a partir dos métodos de implementação do livro-texto do professor Nivio Ziviani. Listas são úteis em aplicações tais como manipulação simbólica, gerência de memória, simulação e compiladores. Na manipulação simbólica os termos de uma fórmula podem crescer sem limites. (ZIVIANI, 1999).

2. DESENVOLVIMENTO

No desenvolvimento do software, foi utilizado o código de implementação de listas do livro-texto do Ziviani, que inclui a definição de listas, e as seguintes funções: uma para determinar se a lista está vazia, uma para esvaziar a lista e uma para inserir um elemento na lista. Além disso, o código do Ziviani, na sua definição de lista, já trazia a implementação do tipo Item, que foi modificado para atender as demandas do projeto, com a inclusão do tipo Recorrência, que armazena a repetição de cada cor na pista, e o tipo Ponto Médio, que armazena o meio da pista.

FIGURA 1 - Linha de código da implementação do tipo Item, tipo de dado usado no software.

```
typedef struct {
  TipoChave Chave; // valor do bloco
  TipoRecorencia Quant; // quantidade de itens por bloco
  TipoPontoMedio relativo;//em relação ao conjunto

} TipoItem;
```

A partir de uma imagem fornecida, as cores de cada linha de pixels foram armazenadas em um *array* do tipo lista, por meio da função do Ziviani. Em seguida, criou-se uma função que agrupava as cores de pixel fornecidas no arquivo em sequência dentro de um vetor do tipo Item, nominado Map.

FIGURA 2 - Implementação da função que agrupa as cores de cada linha da imagem.

```
void findRecorrencia(int tam, TipoLista *lista, TipoItem *Map, int *max){
  /*
  Essa função agurupa a informação de recorrênicia em sequência
  de um valor.
  Ex: Na sequência: 0 0 0 0 255 255 255 255 0 0 0 0
  Serão condensados em apenas três casas, 0 225 0, cada
  unidade de informação conterá a quantidade de repetições,
  assim, cada valor chave, será realocado para um outro vetor,
  permitindo que se faça a busca pelo caminho, de forma mais fácil
  */

  int aux = 0, k = 0;
  int cont = 0, pos = 0;
  for(int i = 0; i<tam; i=k){
    aux = lista->Item[i].Chave;
     k = i; cont = 0;
    while(aux == lista->Item[k].Chave){
      cont++; k++;
    }
    lista->Item[i].Quant = cont;
    Map[pos] = lista->Item[i];
    //printf("O valor %d possui %d repetições\n", Map[pos].Chave, Map[pos].Quant);
    pos++;

  }

  *max = pos;

}
```

Dessa forma, foi possível, por meio de uma função que busca a sequência de cores **0 255 128 255 0** dentro do array Map, determinar a existência ou não de uma pista naquela linha da imagem. Caso confirmada a existência de uma pista, seguindo o padrão de cores determinado, era calculado o meio da pista de rolamento, que consiste no ponto médio do *array* representado pela cor **128**, que seria equivalente identificar a faixa central de uma estrada.

FIGURA 3 - Pontos médios de cada linha da pista.

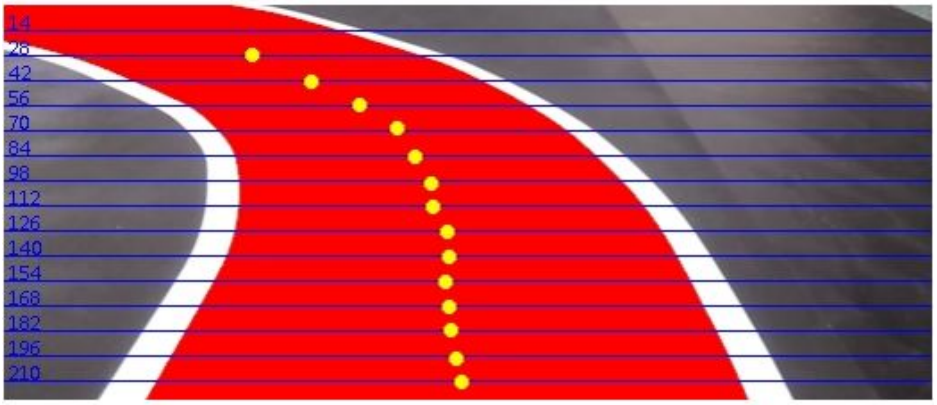

FIGURA 4 - Função que encontra e armazena os pontos médios da pista.

```
void encontraMeio(TipoItem *Map, int pos){
  int count = 0;
  //A função soma, da esquesda para a direita, o intervalo até uma casa antes da pista vermelha.
  //Após isso, ela soma com o valor obtido, com a metade da extensão da pista.
  for(int i = 0; i<=pos; i++){
    if(i == pos){
      if(Map[i].Quant%2 != 0 ){//Aqui, ocorre o arredondamento para cima, quando necessário.
        Map[i].Quant = Map[i].Quant + 1;//Arrendonda para cima
      }
      Map[i].relativo = (count-1)+(Map[i].Quant/2);
    }else
    {
      count+=Map[i].Quant;
    }

  }
}
```

Por fim, com o conhecimento dos pontos de médios de cada linha da imagem foi possível determinar a direção da pista, isto é, se ela virava para direita, ou para a esquerda ou se seguia em linha reta. Para esse último caso em particular, foi considerado uma margem de variação de 20% do ponto médio, pois, pequenos deslocamentos do ponto médio poderiam induzir erroneamente que haveria uma mudança no sentido da pista.

FIGURA 5 - Função que determina o sentido da pista.

```
void verifica_direcao(int v_pmedio[], int *direc, int k){
  k--;
  if(v_pmedio[k] <= (v_pmedio[0]*1.2) && v_pmedio[k] >= (v_pmedio[0]*0.8)){//linha reta.
    *direc = 1;
  }else if(v_pmedio[0] < v_pmedio[k]){//Curva à esquerda.
    *direc = 2;
  }else
    *direc = 3;// Curva à direita.

}
```

3. CONCLUSÃO

De maneira geral, o objetivo geral do projeto foi alcançado, isto é, o aprendizado da implementação de estruturas de dados em simulações. Ademais, sem maiores dificuldades, conseguimos desenvolver um simulador que não apresentasse falhas. Entretanto, o maior desafio encontrado durante o desenvolvimento do software foi a manipulação de casos testes, cujo conteúdo não era disponibilizado ou apresentava resultados ambíguos.

4. REFERÊNCIAS

ZIVIANI, Nivio. **Projeto de Algoritmos Com Implementações em Pascal e C IV edição**, Ed. Pioneira, São Paulo, 1999. Deisponível em: clip2net.com/clip/m1076/1206625061-ebook-projetos-de-algoritmos-com-implementazhes-em-pascal-e-c-nivio-ziviani-4ed-3618kb.pdf. Acessado em: 24 nov. 2020.

ZIVIANI, Nivio. **Projeto de Algoritmos Com Implementações em Pascal e C**, Ed. Cengage, Belo Horizonte. Deisponível em: www2.dcc.ufmg.br/livros/algoritmos/cap3/codigo/c/3.1a3.2-lista-arranjo.c. Acessado em: 24 nov. 2020.