

Passo a passo para ler uma matriz de pixel de uma imagem *.ppm usando alocação de memória e salvar a mesma matriz em novo arquivo *.ppm.

```
#include <stdio.h>
#include <stdlib.h>
```

```
struct Pixel
{
    int R, G, B;
};
```

```
int main ()
{
    FILE *fp;
    int c;
    char tipolmg[3];
    int i, j, linhas, colunas, valor, r, g, b;
```

```
    fp = fopen("golden_novo.ppm","r");
```

```
    // Lê o cabeçalho do arquivo
```

```
    fscanf(fp, "%s", tipolmg);
```

```
    fscanf(fp, "%d %d", &colunas, &linhas);
```

```
    fscanf(fp, "%d", &valor);
```

```
    // lê o tipo de imagem P3
```

```
    // lê o tamanho da matriz
```

```
    // lê o valor máximo.
```

// Alocação da matriz de pixels RGB

```
struct Pixel **matriz = (struct Pixel **)malloc(linhas * sizeof(struct Pixel *));  
for (i = 0; i < linhas; i++)  
{  
    matriz[i] = (struct Pixel *)malloc(colunas * sizeof(struct Pixel));  
}
```

// Leitura dos valores de pixel RGB

```
for (i = 0; i < linhas; i++)  
{  
    for (j = 0; j < colunas; j++)  
    {  
        fscanf(fp, "%d %d %d", &r, &g, &b);  
        matriz[i][j].R = r;  
        matriz[i][j].G = g;  
        matriz[i][j].B = b;  
    }  
}
```

// Fecha o arquivo.

```
fclose(fp);
```

// Criação do novo arquivo PPM

```
FILE* fp_novo = fopen("golden_igual.ppm", "w");  
if (fp_novo == NULL) {  
    printf("Erro ao criar o arquivo.\n");  
    exit(1);  
}
```

// Escreve o cabeçalho do arquivo PPM

```
fprintf(fp_novo, "P3\n");  
fprintf(fp_novo, "%d %d\n", colunas, linhas);  
fprintf(fp_novo, "%d\n", valor);
```

// Escreve os valores RGB da matriz no arquivo

```
for (i = 0; i < linhas; i++)  
{  
    for (j = 0; j < colunas; j++)  
    {  
        fprintf(fp_novo, "%d %d %d ", matriz[i][j].R, matriz[i][j].G, matriz[i][j].B);  
    }  
    fprintf(fp_novo, "\n");  
}
```

// Fecha o arquivo

```
fclose(fp_novo);
```

```
// Libera a memória alocada para a matriz de pixels RGB  
for (i = 0; i < linhas; i++)  
{  
    free(matriz[i]);  
}  
free(matriz);  
  
return 0;  
}
```