#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ncurses.h>

void global\_creationEtRemplissageTbl (int v,int w);

int main(void)

{

char caractereActuel;

char largeur[3];

char hauteur[4];

int largeurInt;

int hauteurInt;

FILE\* fichier = fopen("image\_PBM6", "r");

if (fichier != NULL)

{

// Boucle de lecture des caractères un par un

do

{

caractereActuel = fgetc(fichier); // On va lire le caractère

}

while (caractereActuel != '\n'); // On continue tant que fgetc n'a pas retourné un saut a la ligne

fgets (largeur,3,fichier);

fgets (hauteur,4,fichier);

largeurInt = atoi (largeur)+1;

hauteurInt = atoi (hauteur)+1;

printf("%d&%d\n",largeurInt,hauteurInt );

fclose(fichier);

global\_creationEtRemplissageTbl(largeurInt, hauteurInt);

}

return 0;

}

void global\_creationEtRemplissageTbl (int largeur,int hauteur)

{

char \*\*tab;

char caractereActuel = 'a';

char caractereImage = 'W';

int blanc = 219;

int noir = 255;

FILE \*fichier = fopen("image\_PBM6", "r");

tab = malloc((hauteur) \* sizeof(char\*)); // allocation en mémoire du tableau

for (int i = 0 ; i < largeur ; i++)

{

tab[i] = malloc((largeur) \* sizeof(char));

}

for (int i = 0; i < hauteur; ++i)

{

for (int j = 0; j < largeur; ++j)

{

tab[i][j]=0;

}

}

for (int a = 0; a < 2; ++a)

{

do

{

caractereActuel = fgetc(fichier); // On lit le caractère

} while (caractereActuel != '\n'); // On continue tant que fgetc n'a pas retourné un saut de ligne

}

for (int i = 0; i < hauteur; ++i) // écriture tableau

{

for (int j = 0; j < largeur; ++j)

{

caractereImage = fgetc (fichier);

switch (caractereImage){

case '0':

tab[i][j]= noir;

break;

case '1':

tab[i][j]= blanc;

break;

case ' ':

j--;

break;

default:

tab[i][j]=' ';

break;

}

}

}

fclose(fichier);

initscr();

int posline = LINES/2 - hauteur/2; //Position curseur

int poscol = COLS/2 - largeur/2;

move(posline,poscol);

for (int i = 0;i < hauteur; ++i) // affichage tableau

{

for (int j = 0;j < largeur; ++j)

{

printw("%c", tab[i][j-1]);

move(posline,poscol);

poscol ++

}

posline ++;

poscol = COLS/2 - largeur/2;

move(0,0);

}

refresh();

getch();

endwin();

}