#include <iostream>

#include <conio.h>

#include <windows.h>

#include <fstream>

#include <string>

using namespace std;

/\*This is my main Node class where it can

hold a char value and pointers to the next and previous node\*/

class Node

{

public:

char c;

Node\* next;

Node\* prev;

};

/\*function used to manipulate the cursor\*/

void gotoxy(int x, int y)

{

COORD pos = { x, y };

HANDLE output = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleCursorPosition(output, pos);

}

/\*function that creates the link list includes:

flags and counters for copy and paste functions,

Node pointers to help manipulate the list...etc\*/

void createNew()

{

char copy[10];

int copycount = 0,ycount =0;

Node\* arr[100] = { nullptr };

Node\* start = nullptr;

Node\* end = nullptr;

Node\* prevNode = nullptr;

int xcur = 0, ycur = 0, flag =0,flag2 = 0, max =0,color =4;

ifstream indoc;

string fname;

char file[sizeof(indoc)];

int j = 0;

char letter = ' ';

while (letter != 27)

{

letter = \_getch();

if (letter == 8) // backspace

{

Node \*p;

p = end;

if (p != start) //if there is no prev node, dont delete

{

end = end->prev;

delete p;

end->next = nullptr;

xcur--;

}

}

else if (letter == -32)

{

// he pressed right up left or down

letter = \_getch();

//pressed left, decrement xcur, move cursor back

if (letter == 75)//72 is up 75 left 77 right 80 down

{

--xcur;

end = end->prev;

}

//pressed right, increment xcur, move cursor forward

if (letter == 77)

{

xcur++;

end = end->next;

}

if (letter == 80) //pressed down, moved down

ycur++;

if (letter == 72) //presed up, move up, move current cursor

{

ycur--;

xcur--;

end = arr[ycur];

}

}

else if (letter == 95) //copy '\_'

{

for (int i = 0; i <= ycount; i++) //if there are multiples rows of text

{

Node \*s;

s = arr[i];

while (s != nullptr) //loops through list, coppies text data to an array

{

copy[copycount] = s->c;

copycount++;

s = s->next;

max = copycount;

}

}

}

else if (letter == 45) //paste - increments flag to signal to paste the copied data

{

flag2++;

}

else if (letter == 43) //save press '+'

{

//creates a output file for saving

ofstream outdoc;

outdoc.open("TheEditor.txt");

Node\* save;

char data;

save = start;

while (save->next != nullptr) //prints data to file

{

data = save->c;

outdoc << data;

save = save->next;

}

outdoc.close();

}

else if (letter == 61) //load file '='

{

//creates input file for loading a file and asks for name of file

cout << "\nPlease enter the name of the file" << endl;

cin >> fname;

system("cls");

indoc.open(fname);

while (!indoc.eof()) //loading the saved file into an array

{

indoc >> file[j];

j++;

}

j = j-2;

for (int i = 0; i <= j; i++) //printing the file

{

cout << file[i];

}

flag++;

}

//If the user presses enter, a new list it started on a new line

//of a 2D array

else if (letter == 13)

{

++ycur;

arr[ycur] = nullptr;

start = arr[ycur];

xcur = 0;

ycount++;

copycount = 0;

}

//normal letter is entered, a linked list is started

else

{

Node\* p = new Node();

p->c = letter;

p->next = nullptr;

p->prev = nullptr;

if (start == nullptr)

{

start = p;

end = p;

prevNode = p;

arr[ycur] = start;

}

// insert in middle if the current cursor position is not

//at the end of the list, this function will insert it

else if (end->next != nullptr)

{

p->prev = end;

p->next = end->next;

end->next = p;

prevNode = p;

end = p;

}

//Adding more nodes to the end

else

{

p->prev = prevNode;

p->next = nullptr;

end->next = p;

end = p;

prevNode = p;

}

//incrementing the color scheme and the xposition

xcur++;

color++;

}

system("cls");

// display your linked list here

if (flag == 1) //If a file is loaded this nested loop will print it

{

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

{

cout << file[i];

}

}

else if (flag2 == 1) //If a part of the text is copied then pasted, this will print it

{

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

{

SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), color);

cout << copy[i];

}

}

for (int i = 0; i <= ycount; i++) //Displaying normal link list

{

Node \*s;

s = arr[i];

while (s != nullptr)

{

SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), color);

\_putch(s->c);

s = s->next;

}

cout << endl;

}

/\*These are for the cursor position, changes when

the user either uploads a file or copy and

pastes text\*/

if (flag == 1 && flag2 == 1) //If loaded in file and copy

{

gotoxy(xcur + j + copycount, ycur);

}

else if (flag2 == 1) //if file is only copied and pasted

{

gotoxy(xcur + copycount, ycur);

}

else if (flag == 1) //if a file is just loaded un

{

gotoxy(xcur + j, ycur);

}

else

gotoxy(xcur, ycur); //else a file is not loaded and text is not copied and pasted

indoc.close();

}

}

int main()

{

SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), 30);

cout << "Welcome to theEditor!\n" //Main Menue

<< "Controls:\n"

<< "1. Load File '='\n"

<< "2. Save File '+'\n"

<< "3. Copy '\_'\n"

<< "4. Paste '-'" << endl;

system("pause");

system("CLS");

createNew();

return 0;

}