

Windows NotePad

윈도우의 메모장기능을 만들어보자.

기능은 아래와 같다.

1. 텍스트 입력
2. 커서 이동
3. Redo

텍스트를 입력하면 커서로부터 입력된다. 커서는 가장 왼쪽이 1이다. 함수설계는 아래와 같다

| 함수명 | 기능 |
|---|---|
| void init(); | 전체삭제하고 모두 초기화한다. |
| void input_txt(char* ch); | 커서로부터 텍스트를 삽입한다. 뒷부분에 텍스트가 있다면 밀려난다. |
| void move_cursor(int n); | 커서를 이동한다 가장 왼쪽이 1이며, 1보다작은경우는 입력되지않는다. 가장오른쪽 보다 클 경우 텍스트가 있는 가장 오른쪽으로 가면 된다. |
| void redo(); | 이전 연속적으로 작성된 텍스트가 사라지고 사라진 텍스트의 시작위치로 커서가 이동한다. 다시한번 호출한다면 텍스트를 다시 원래대로 되돌리고 되돌린 텍스트의 마지막으로 커서가 간다. |
| void check(int start, int length, char* str); | start부터 length길이만큼 str에 복사한다. str배열의 끝에 0을 꼭 입력해준다. 가장 왼쪽끝의 start값은 0이다. |

아래 그림을 참고

| | |
|--------------|---------------|
| 1. abc입력 | a b c |
| 2. 커서2로이동 | a b c |
| 3. ddd입력 | a d d d b c |
| 4. 커서 3으로이동 | a d d d b c |
| 5. 커서 10으로이동 | a d d d b c |
| 6. redo호출 | a b c |
| 7. redo호출 | a d d d b c |

| input | output |
|-------|--------|
| 1 | bcc |
| 2 aa | ab |
| 2 bb | cc |
| 2 cc | |
| 4 | ab |
| 4 | |
| 2 dd | |
| 3 2 | |
| 2 ff | |
| 5 5 3 | |
| 1 | |
| 2 aa | |
| 2 bb | |
| 3 2 | |
| 2 cc | |
| 4 | |
| 5 1 2 | |
| 4 | |
| 5 1 2 | |
| 1 | |

| | |
|---|--|
| 2 aa 2 bb 3 2 3 5 2 cc 4 5 1 2 4 5 1 2 6 | |
|---|--|

| |
|-----------|
| Main code |
|-----------|

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

using namespace std;

void init()
{

}

void input_txt(char ch[])
{

}

void move_cursor(int n)
{

}

void redo()
{

}

void check(int start, int len, char str[])
{
```

```

}

int main()
{
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    int order, n, st;
    char str[32];
    while (1){
        scanf("%d", &order);
        switch (order){
            case 1:
                init();
                break;
            case 2:
                scanf("%s", str);
                input_txt(str);
                break;
            case 3:
                scanf("%d", &n);
                move_cursor(n);
                break;
            case 4:
                redo();
                break;
            case 5:
                scanf("%d %d", &st, &n);
                check(st, n, str);
                printf("%s\n", str);
                break;
            default:
                return 0;
        }
    }
}

```