



{ 목 차 }

▶ 브레인스토밍	03
▶ 구상 동기	04
▶ 자료조사	05
▶ 역할 분담	06
▶ 소스 제작	07

" Music begins where the language ends. – Mozart"

{브레인 스토밍}

▶ "머리 속 폭풍: 혁신적인 아이디어를 찾아서"

1.
냉장고 안에 뭐가 들었을까요?



냉장고 안에 어떤 물건이 들었는지 프로그래밍을 통해 알려주는 프로그램

2.
버스노선/지하철노선 알려주기



버스나 지하철의 도착시간, 배차간격 등을 알려주고, 어떤 노선인지 설명해주는 프로그램

3.
스무고개



랜덤의 단어를 프로그램이 임의로 지정한 다음 그 단어를 스무고개 형식으로 맞추는 프로그램

{브레인 스토밍}

▶ "머리 속 폭풍: 혁신적인 아이디어를 찾아서"

4.

주차장 요금계산 프로그램



주차장에 어떤 차가 들어가고 들어간 시간을 파악해서 요금을 계산해주는 프로그램

5.

식물을 키우는 프로그램



식물의 종류마다 어떤 주기로 물을 주고 어떻게 하면 식물이 잘 크는지에 대한 프로그램

6.

피아노 프로그램



키보드마다 함수를 부여해 누르면 음계가 출력되고 저장되는 프로그램

{ 구상동기 }

▶ 주차요금 프로그램

평소에 많이 보기 때문에 그에 대한 흥미가 생겼고 알고리즘이 복잡하지 않고 포인터와 구조체 활용이 유용할 것 같아 이 주제를 선택하였지만, 생각보다 단순한 프로그램이 될 것 같아서 선택하지 않았다

▶ 냉장고 목록보기

콘솔프로그램을 짜야 하는 프로젝트인데 센서의 필요성이 있을 것 같아서 선택하지 않음

▶ 스무고개 프로그램

스무고개는 너무 흔한 것 같아서 선택하지 않음

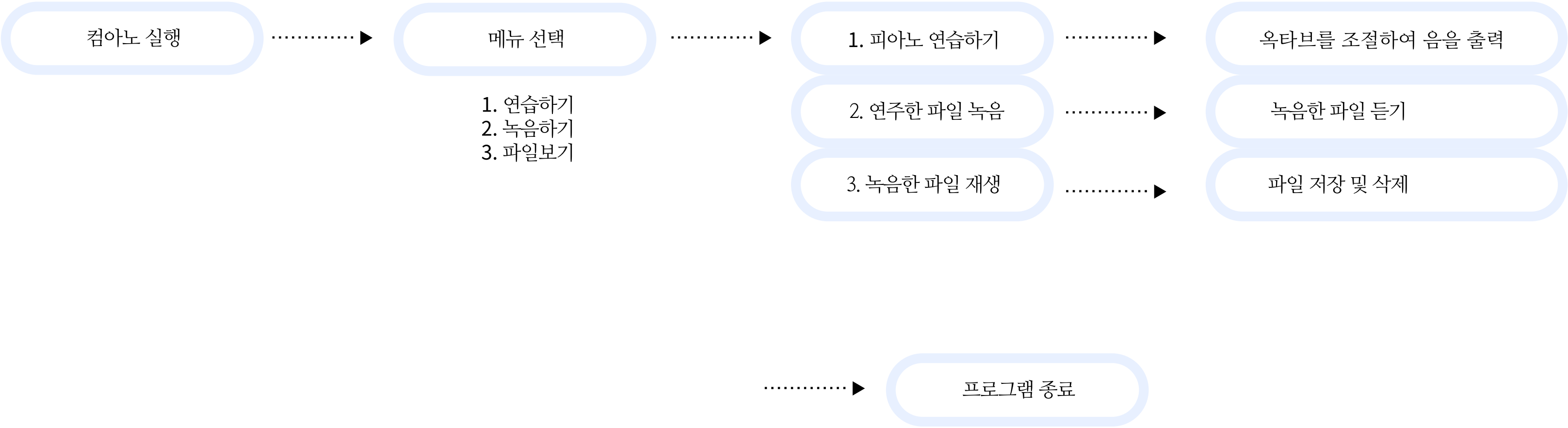
▶ 식물키우기 프로그램

식물을 키우는 데에 사소한 문제들을 해결할 수가 없었다

▶ 피아노 연주 프로그램

키보드로 피아노를 연주한다는 신선한 주제라서 선택하게 되었다

{ 알고리즘 }



{ 전략, 역할 분담 }

▶ PPT 제작

..... ●

최종 발표 및 매주 프로젝트 진행보고서를 위한 PPT 제작

▶ 자료조사

..... ●

컴퓨터로 피아노를 구현하기 위해
피아노 음과 비슷한 소리를 내는 주
파수등에 대한 정보를 수집

▶ 소스 제작

..... ●

실제 구현을 위한 소스 제작

▶ 브레인스토밍

..... ●

컴아노를 만들기 위한 아이디어를
도출하기 위한 과정

{ 자료 조사 }

전체 이미지 동영상 쇼핑 뉴스 더보기

도구

검색결과 약 72,400개 (0.26초)

달나라 노트
<https://cosmosproject.tistory.com> › ...

Piano 음계별 주파수(Hz) - 달나라 노트

2021. 3. 30. — 1, 2, 3, 4, 5, 6, 7, 8. 1, 도, C, 16.35, 32.70, 65.41, 130.81, 261.63, 523.25, 1046.50, 2093.00, 4186.01. 2, 도#, C#, 17.32, 34.65, 69.30 ...

Naver Blog
<https://blog.naver.com> › ylefor

헤르츠 / 헤르쯔 / 피아노 / 피아노음계 / 주파수 / 피아노 건반수 ...

2014. 7. 9. — ... 음의 최대 가청범위는 20hz - 2만hz라고 하는데 피아노의 88건반은 최저음이 27.5hz, 최고음 이 4186hz이며 소리로는 그 4배 음까지 들을 수 있으므로 ...

Naver Blog
<http://blog.naver.com> › ilesson

[피아노레슨] 피아노 건반의 주파수

2012. 5. 23. — C 건반은 주파수가 4,186.0091 Hz입니다. 피아노 건반의 기준음은 49번 A 로 440Hz 입니다. 거의 모든 피아노는 440Hz를 기준으로 조율되어 있습니다.

			N								
K	notes	notes	0	1	2	3	4	5	6	7	8
1	도	C	16.35	32.70	65.41	130.81	261.63	523.25	1046.50	2093.00	4186.01
2	도#	C#	17.32	34.65	69.30	138.59	277.18	554.37	1108.73	2217.46	4434.92
3	레	D	18.35	36.71	73.42	146.83	293.66	587.33	1174.66	2349.32	4698.64
4	레#	D#	19.45	38.89	77.78	155.56	311.13	622.25	1244.51	2489.02	4978.03
5	미	E	20.60	41.20	82.41	164.81	329.63	659.26	1318.51	2637.02	5274.04
6	파	F	21.83	43.65	87.31	174.61	349.23	698.46	1396.91	2793.83	5587.65
7	파#	F#	23.12	46.25	92.50	185.00	369.99	739.99	1479.98	2959.96	5919.91
8	솔	G	24.50	49.00	98.00	196.00	392.00	783.99	1567.98	3135.96	6271.93
9	솔#	G#	25.96	51.91	103.83	207.65	415.30	830.61	1661.22	3322.44	6644.88
10	라	A	27.50	55.00	110.00	220.00	440.00	880.00	1760.00	3520.00	7040.00
11	라#	A#	29.14	58.27	116.54	233.08	466.16	932.33	1864.66	3729.31	7458.62
12	시	B	30.87	61.74	123.47	246.94	493.88	987.77	1975.53	3951.07	7902.13

(단위 : Hz)

피아노는 보통 8옥타브로 구성되어있고, 각 계이름(note)별 주파수는 위와 같습니다.

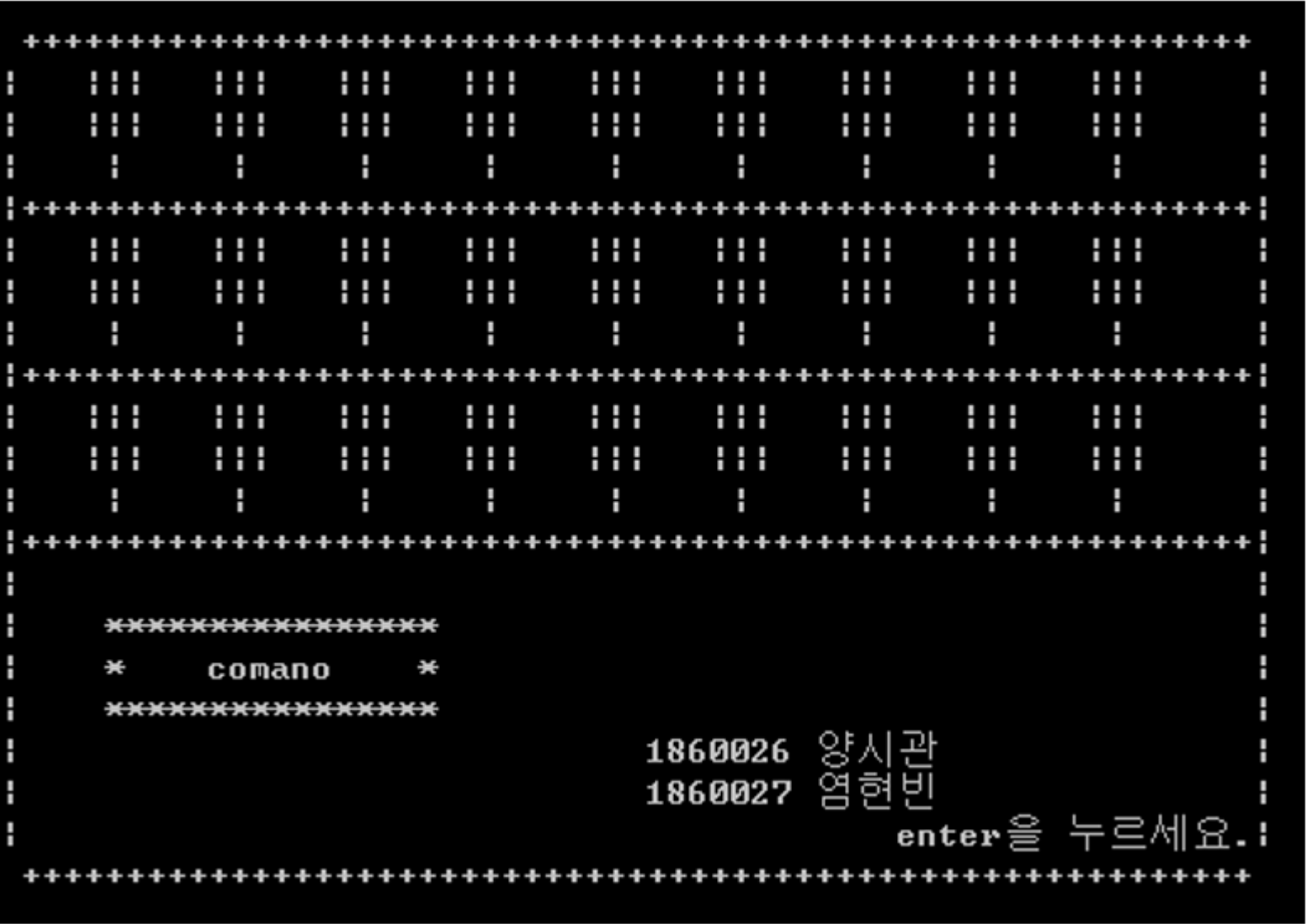
피아노 조율을 할 때도 위 내용이 사용되지요.

{ 소스 제작 }

```
int main(void)
{
    intro();
    display_manual();
    manual();
    display_piano_keyboard();
    getchar();
    getchar();
    return 0;
}

void intro() {
    printf("\n");
    printf("\n");
    printf("+++++\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("+++++\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("+++++\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("+++++\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("|   |   |   |   |   |   |   |   |   |   |\n");
    printf("+++++\n");
    printf("      * comano *\n");
    printf("      * comano *\n");
    printf("      * comano *\n");
    printf("1860026 양시관\n");
    printf("1860027 염현빈\n");
    printf("enter을 누르세요.\n");
    printf("+++++\n");

    getchar();
    system("cls");
}
```



{ 소스 제작 }

```
//피아노 모양 꾸미기 함수
void display_piano_keyboard(void)
{
    int i;
    printf("1부터 8까지 숫자 키를 누르면\n");
    printf("각음의 소리가 출력됩니다\n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  |  _  _  _  _  _  _  _  _  _  _  \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    printf("  | | | | | | | | | | | | | | | | \n");
    char code[8][4] = { "도", "레", "미", "파", "솔", "라", "시", "도" };
    //피아노 안에 키보드 키 표시
    for (i = 0; i < 8; i++)
    {
        gotoxy(3 + i * 4, 8);
        printf("%2d", i + 1);
    }
    //피아노 안에 각 피아노의 음을 표시
    for (i = 0; i < 8; i++)
    {
        gotoxy(3 + i * 4, 10);
        printf("%s", code[i]);
    }
    printf("\n");
    printf("\n\n\n-----\n");
}
```

1부터 8까지 숫자 키를 누르면
각음의 소리가 출력됩니다

1	2	3	4	5	6	7	8
도	레	미	파	솔	라	시	도

{ 소스 제작 }

```
//프로그램 간단설명서
void display_manual(void)
{
    printf("\n");
    printf("키보드를 누르면\n");
    printf("키보드에 음이 표시됩니다.\n");
    printf("직접 음을 연주해보세요.\n");
    printf("enter를 누르세요");
    printf("\n");

    getchar();
    system("cls");

    return 0;
}
```

```
//프로그램 메뉴얼 안내
//프로그램 목차 설명
void manual(void)
{
    int title;
    printf("0). 악보보기\n");
    printf("1). 연습하기\n");
    printf("2). 녹음하기\n");
    printf("3). 파일보기\n");
    printf("4). 종료\n");
    printf("5). 메인화면으로 돌아가기\n");

    scanf("%d", &title);
    system("cls");
}
```

D:\유템프로젝트\Debug\유템프로젝트.exe

키보드를 누르면
키보드에 음이 표시됩니다.
직접 음을 연주해보세요.
enter를 누르세요

0). 악보보기
1). 연습하기
2). 녹음하기
3). 파일보기
4). 종료
5). 메인화면으로 돌아가기

{ 소스 제작 }

//악보 종류 선택하는 함수

void sheet(void)

{

int n = 0;

```
printf("1.%s\n", song_name[0]); //비행기
printf("2.%s\n", song_name[1]); //작은별
printf("3.%s\n", song_name[2]); //나비야
printf("4.%s\n", song_name[3]); //엄마가 섬그늘에
printf("5.%s\n", song_name[4]); //풍선
printf("6.%s\n", song_name[5]); //곰세마리
printf("7.%s\n", song_name[6]); //옹달샘
printf("8.%s\n", song_name[7]); //산토끼
printf("9.%s\n", song_name[8]); //강아지
printf("10.%s\n", song_name[9]); //자전거
printf("11.%s\n", song_name[10]); //개구리
printf("12.%s\n", song_name[11]); //둥글게 둥글게
printf("13.%s\n", song_name[12]); //둥근해가 떴습니다
printf("14.%s\n", song_name[13]); //거미가 줄을타고 올라갑니다
printf("15.%s\n", song_name[14]); //머리어깨무릎발
printf("16.메인화면으로 돌아가기\n");
```

```
scanf("%d", &n);
system("cls");
```

D:\#템프로젝트#Debug#템프로젝트.exe

```
1.비행기
2.작은별
3.나비야
4.엄마가 섬그늘에
5.풍선
6.곰세마리
7.옹달샘
8.산토끼
9.강아지
10.자전거
11.개구리
12.둥글게 둥글게
13.둥근해가 떴습니다
14.거미가 줄을타고 올라갑니다
15.머리어깨무릎발
16.메인화면으로 돌아가기
```

{ 소스 제작 }

```

if (n == 1)
{
    printf("1.%s\n", song_name[0]);    //비행기
    printf("제이름 번호입니다\n");
    printf("3 2 1 2 3 3 3 2 2 2 3 3 3 2 1 2 3 3 3 2 2 3 2 1\n");
    printf("\n");

    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf(" _ □ _ _ □ □ _ _ □ □ □ _ _ _ _ _ \n");
    printf("  □  □      □ □ □      □ \n");
    printf("    □                               □ \n");

    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf("-----\n");
    printf(" _ _ □ _ □ _ _ _ □ _ _ _ _ _ \n");
    printf(" □      □ □      □ \n");
    printf("                               □ \n");

    system("pause");
    system("cls");
    sheet();
}

```

[illegible]

{ 소스 제작 }

```
//피아노 연습하기
void practice_piano(void)
{
    int index[] = { 0, 2, 4, 5, 7, 9, 11, 12 };
    int freq[8], code, i;

    for (i = 0; i < 8; i++)
    {
        freq[i] = calc_frequency(4, index[i]);
    }

    do
    {
        code = _getch();
        if ('1' <= code && code <= '8')
        {
            code -= 49;
            Beep(freq[code], 200);
        }
    } while (code != 27);
}
```



{ 소스 제작 }

```
//피아노 연주한 노래를 녹음하는 함수
void record_piano(void)
{
    //오르프 끝내고 녹음한다고 출력하는 함수
    char c;
    int index[] = { 0, 2, 4, 6, 7, 9, 11, 12 };
    int freq[8], code, i;
    int j = 0;
    int a[100];
    int count = 0;
    for (i = 0; i < 8; i++)
    {
        freq[i] = calc_frequency(4, index[i]);
    }

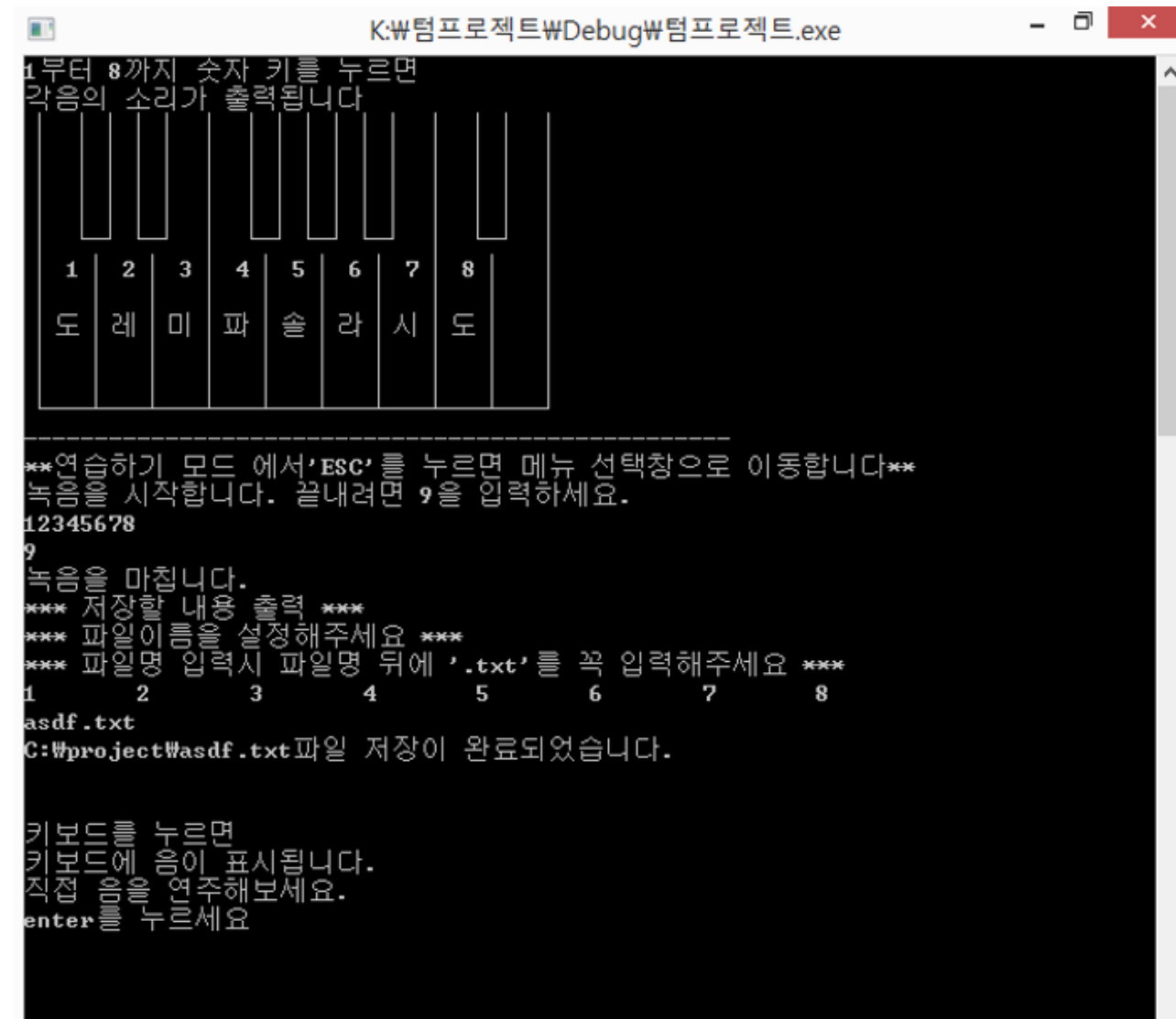
    printf("녹음을 시작합니다. 끝내려면 9를 입력하세요.\n"); // 화면에 녹음 시작을 출력
    while (j < 100)
    {
        c = getch(); //키보드로 한글자 입력받기, scanf함수인 경우 엔터키를 입력해야함
        code = c - 48; //입력받은 키 값을 숫자로 변경하기 위해 48을 뺌(숫자 0의 아스키코드값=48)
        Beep(freq[code - 1], 200);
        if (code == 9)
        {
            printf("녹음을 마칩니다.\n");
            break;
        }
        else if (1 <= code && code < 9) //code != ('\n' - 49)
        {
            a[j] = code;
            count++;
            j++;
        }
    }

    printf("*** 저장할 내용 출력 ***\n"); // 화면에 녹음한 음을 출력
    printf("*** 파일이름을 설정해주세요 ***\n");
    printf("*** 파일명 입력시 파일명 뒤에 '.txt'를 꼭 입력해주세요 ***\n");
    //파일명 저장 관련 설명
    //반드시 .txt를 붙여줘야 텍스트파일로 저장됨
    //.txt로 저장되면 3)파일보기에서 텍스트형식의 파일을 볼 수 있음

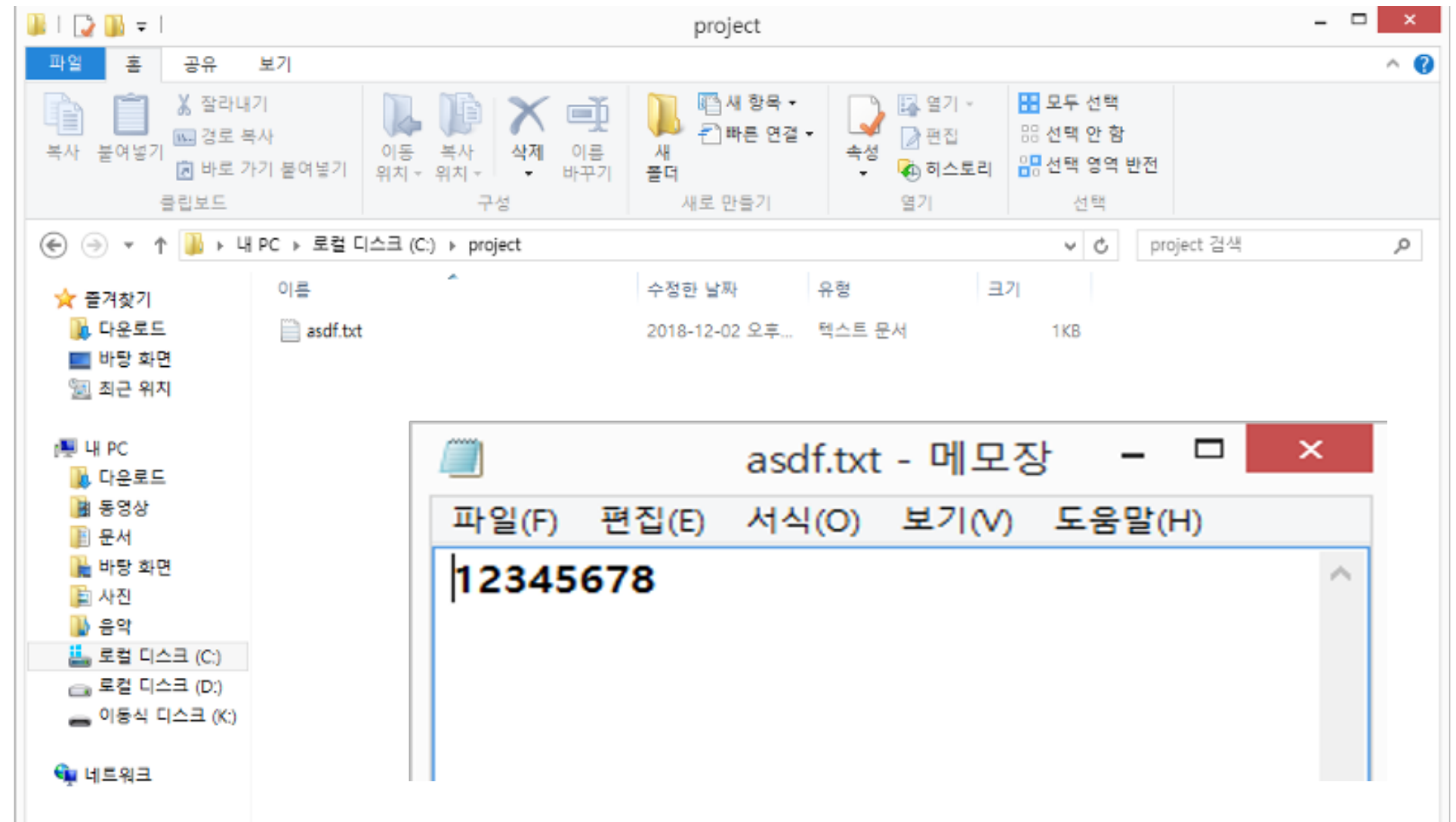
    for (j = 0; j < count; j++)
    {
        printf("%d #t", a[j]);
    }

    save_record_data(a, count); // 입력된 내용을 저장

    getch();
}
```



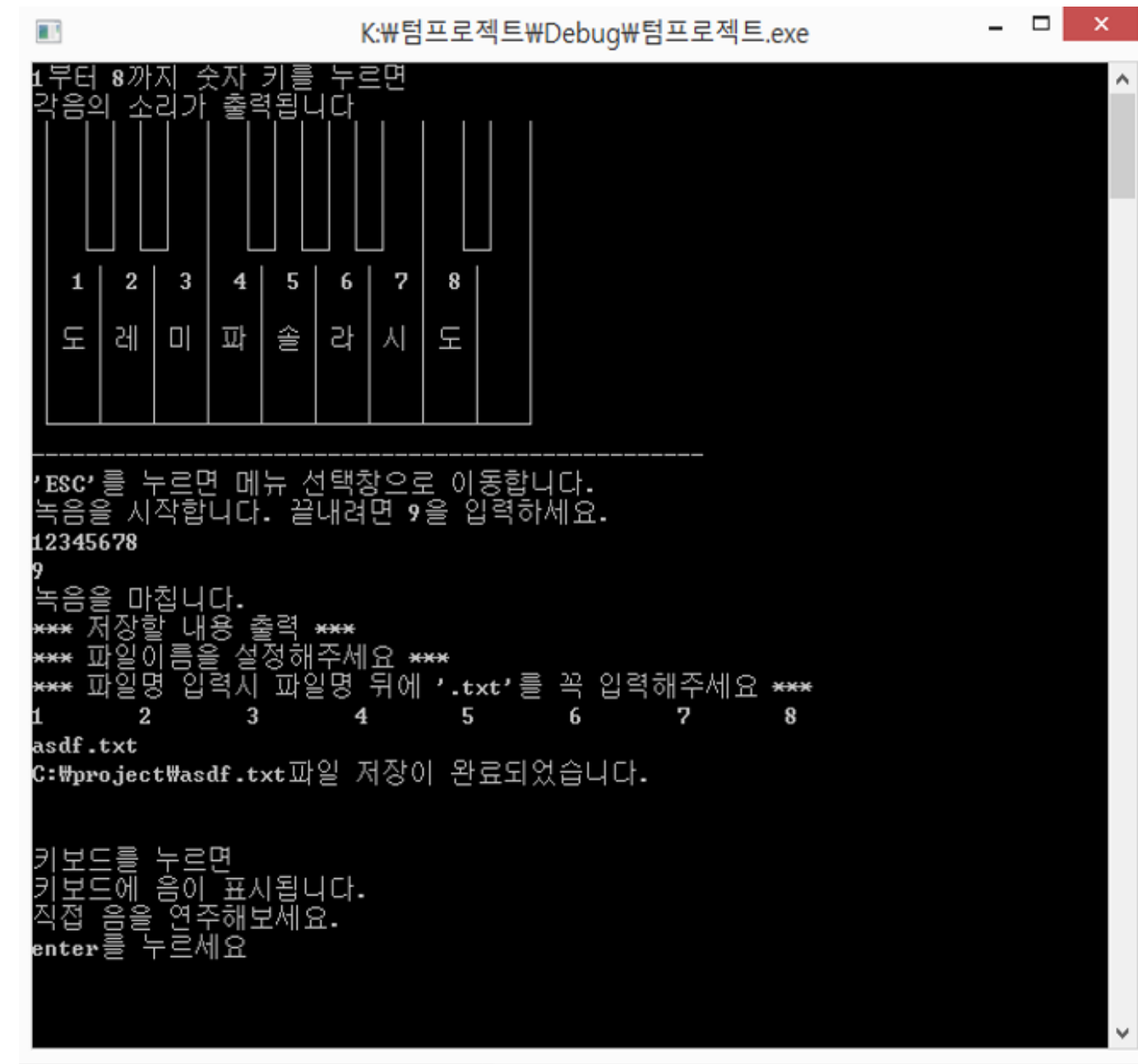
{ 소스 제작 }



{ 소스 제작 }

```
//연주한 노래를 데이터로 저장하는 함수
void save_record_data(int arr[], int arrsize)
{
    int idx = 0;
    char filename[100];
    char path[255] = "C:\\\\project\\\\";
    //C:\\project에 저장
    //C드라이브에 project라는 폴더를 처음으로 만들어줌

    scanf("%s", filename);
    strcat(path, filename);
    printf(path);
    FILE *pFile = NULL;
    pFile = fopen(path, "wb"); // 읽어올일 파일명
    if (pFile != NULL)
    {
        for (idx = 0; idx < arrsize; idx++)
        {
            fprintf(pFile, "%d\\n", arr[idx]);
        }
        fclose(pFile); // 파일 종료
        printf("파일 저장이 완료되었습니다.\\n\\n");
    }
    else
    {
        printf("파일 저장에 실패하였습니다.\\n\\n");
    }
}
```



{ 소스 제작 }

```
//녹음한 파일을 보여주기
void see_record(void)
{
    FILE *pFile = NULL;
    int* buff;
    int data = 0;
    int read_size = 0;           // 읽어들이는 데이터 양
    char filename[100];
    char path[255] = "C:\\project\\"; //저장했던 txt 파일을 불러오는 경로
    scanf("%s", filename);
    strcat(path, filename);
    pFile = fopen(path, "r");    // 읽어들이는 파일명
    if (pFile == NULL)
    {
        printf("err");
        return;
    }
    buff = (int *)malloc(sizeof(int) * 256); // 지정 크기만큼 char 배열을 생성(동적할당)

    int freq[8], code, i;
    int index[] = { 0, 2, 4, 5, 7, 9, 11, 12 };
    for (i = 0; i < 8; i++)
    {
        freq[i] = calc_frequency(4, index[i]);
    }
    while (fscanf(pFile, "%d", &data) != EOF)
    {
        Beep(freq[data - 1], 200);
        printf("%d ", data);
    }
    getchar();
}
```



{ 소스 제작 }

```
#define _CRT_SECURE_NO_WARNINGS
#include<stdio.h>
#include <math.h>
#include <string.h>
#include<windows.h>
#include <conio.h>

struct song
{
    char song_name[300];
};
typedef struct song song;

//노래 제목을 위한 문자열
char *song_name[] = {
    "비행기", "작은별", "나비야", "엄마가 섬그늘에", "풍선", "곰세마리", "옹달샘", "산토끼",
    "강아지", "자전거", "개구리", "둥글게 둥글게", "둥근혜가 뗏습니다", "거미가 줄을타고 올라갑니다", "머리어깨무릎발"
};

void intro();
void display_manual(void);
void display_piano_keyboard(void);
int calc_frequency(int octave, int inx);
void practice_piano(void);
void record_piano(void);
void see_record(void); //char*filepath
void manual(void);
void gotoxy(int x, int y);
void save_record_data(int arr[], int arrsize);
void anykey(void);
int n;

int main(void)
{
    intro();
    display_manual();
    manual();
    practice_piano();
    record_piano();
    see_record();
    display_piano_keyboard();

    getchar();
    getchar();

    return 0;
}
```

```
//주파수 계산
//소리 출력을 위한 주파수 계산 공식
int calc_frequency(int octave, int inx)
{
    double do_scale = 32.7032;
    //'도'의 음계에 해당하는 주파수 32.7032
    //한 옥타브 위로 올릴 시 주파수 65.4064
    double ratio = pow(2., 1 / 12.);
    int i;
    double temp = do_scale * pow(2., octave - 1);
    for (i = 0; i < inx; i++)
    {
        temp = (int)(temp + 0.5);
        temp *= ratio;
    }
    return (int)temp;
}
```

```
void gotoxy(int x, int y)
{
    COORD Pos = { x - 1, y - 1 };

    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), Pos);
}
```

• A Programmed Piano •
{감사합니다.}

