```c
#include <stdio.h>
#include <string.h>
//Windows API
#include <windows.h>


#include<conio.h>

// For music addition
#include <mmsystem.h>
#pragma comment(lib, "winmm.lib")

//For inventory and Achivements
#define MAX_RESTARTS 3
#define MAX_ITEMS 10
#define strcasecmp _stricmp
#define MAX_ACHIEVEMENTS 11

int achievements[MAX_ACHIEVEMENTS] = {0};
int difficulty_mode = 1; //Foe difficulty mode
int game_paused = 0;  // To pause the timer

//Structure for the timer
typedef struct Timer
{
    int duration;
    volatile int running;
} Timer;

//Tread create event
HANDLE pause_event = NULL;
// TO keep count of thr restart
static int count = 1;

//Keep track of timer and for bonus time and penalty
volatile int gameRunning   = 0;
volatile int timerFinished = 0;

//Loop control for rooms
volatile int exitroom1      = 0;
volatile int exitroom2      = 0;
volatile int exitroom3      = 0;
volatile int exitroom4      = 0;

//To count of the no, of tries you taketo solve the puzzel
int static try;
int static try2;
int static try3;
int static try4;
int static try5;
```

```c
int static try6;
int static try7;
int static try8;
int static code5;
int static code6;
int static code7;
int static code8;
//To go out of the loop of the room
int static exitgame=0;

//To change the music on giving the correct answer
int static musicchange=0;

char inventory[MAX_ITEMS][50];
int  inventory_count = 0;

//struct variable
Timer gameTimer;


// Background music thread
DWORD WINAPI MusicThread(LPVOID lpParam)
{
    int lastMusic = -1;

    while (gameRunning)
    {
        if (musicchange == 0 && lastMusic != 0)
        {
            PlaySound("background.wav", NULL, SND_FILENAME | SND_LOOP | SND_ASYNC);
            lastMusic = 0;
        }
        else if (musicchange == 1 && lastMusic != 1)
        {
            PlaySound("longnight.wav", NULL, SND_FILENAME | SND_LOOP | SND_ASYNC);
            lastMusic = 1;
        }
        else if (musicchange == 2 && lastMusic != 2)
        {
            PlaySound("alone.wav", NULL, SND_FILENAME | SND_LOOP | SND_ASYNC);
            lastMusic = 2;
        }
        else if (musicchange == 3 && lastMusic != 3)
        {
            PlaySound("ambient.wav", NULL, SND_FILENAME | SND_LOOP | SND_ASYNC);
            lastMusic = 3;
        }
        else if (musicchange == 4 && lastMusic != 4)
        {
            PlaySound("calm.wav", NULL, SND_FILENAME | SND_LOOP | SND_ASYNC);
```

```c
            lastMusic = 4;
        }
        else if (musicchange == 5 && lastMusic != 5)
        {
            PlaySound("silence.wav", NULL, SND_FILENAME | SND_LOOP | SND_ASYNC);
            lastMusic = 5;
        }
        else if(musicchange==99 )
        {
            PlaySound(NULL, NULL, 0);
            lastMusic=99;
        }

        Sleep(500);
    }

    PlaySound(NULL, NULL, 0);
    return 0;
}



// Timer thread
DWORD WINAPI TimerThread(LPVOID lpParam)
{
    Timer *timer = (Timer *)lpParam;
    printf("\n");
    while (timer->duration > 0 && timer->running)
    {

        WaitForSingleObject(pause_event, INFINITE);

        if (!timer->running) break;

        printf("\rTime left: %02d:%02d", timer->duration / 60, timer->duration %
60);
        fflush(stdout);

        if (!game_paused)
        {
            Sleep(1000);
            timer->duration--;
        }
    }


    if (gameTimer.duration == 0)
    {
        printf("\rTime left: 00:00\n");
```

```c
        printf("Enter the exit option to go to the restart menu.\n");
    }

    timerFinished = 1;
    gameRunning = 0;
    return 0;
}


// Achievements
void Achievements()
{
    const char *achievement_names[MAX_ACHIEVEMENTS] = {
    "BrainStromer1              : Solved Chamber of Light Without hints.",
    "BrainStromer2              : Solved Echo Library Without hints.    ",
    "BrainStromer3              : Solved Machine Core Without hints.    ",
    "BrainStromer4              : Solved Chiphere Hall Without hints.   ",
    "Beacon of light            : Solved Chamber of light.             ",
    "ShardSeeker                : Acquired the Shard of light.         ",
    "Whispers of the Forgotten : Solved Echo Library.                 ",
    "Heart of the Vault         : Solved Machine Core.                 ",
    "Codebreaker                : Solved Cipher Hall.                  ",
    "Explorer                   : Explored all the vaults.            ",
    "Core Finder                : Acquired the Motherboard.           "};
    printf("\n------------------- Achievements Unlocked ------------------\n");
    for (int i = 0; i < MAX_ACHIEVEMENTS; i++)
    {
        printf("%2d. %s [%s]\n", i + 1, achievement_names[i],
               achievements[i] ? "Unlocked" : "Locked");
    }
    printf("------------------------------------------------------------\n");
}


// Start/restart prompt with correct ordinal suffix and max 3 restarts
int user_input()
{
    if (count > MAX_RESTARTS)
    {
        printf("You have reached the maximum number of restarts (%d). Game
over!\n", MAX_RESTARTS);
        return 0;
    }

    while (1)
    {
        const char *suffix = (count == 1) ? "st"
                           : (count == 2) ? "nd"
                           : (count == 3) ? "rd"
                           : "th";
```

```c
        printf("\nThis is your %d%s try.\n", count, suffix);

        if (count == 1)
            printf("Type 'start' to enter the game or 'exit' to quit:\n");
        else
            printf("Type 'restart' to re-enter the game or 'exit' to quit:\n");

        char input[16];
        scanf("%15s", input);

        if ((count == 1 && strcmp(input, "start") == 0) ||
            (count >  1 && strcmp(input, "restart") == 0))
        {
            printf("You have entered The Vaults of Echoes!!!!\n");
            count++;
            return 1;
        }
        else if (strcmp(input, "exit") == 0)
        {
            printf("You have quit the Game.\n");
            return 0;
        }
        else
        {
            printf("Invalid input. Try again.\n\n");
        }
    }
}


// Inventory management
//function to add ojects to the inventory
void add_to_inventory(const char *item)
{
    if (inventory_count < MAX_ITEMS)
        strcpy(inventory[inventory_count++], item);
}

//function to see what items are stored in the inventory
void show_inventory()
{
    printf("\n--- Inventory ---\n");
    for (int i = 0; i < inventory_count; i++)
        printf("%2d. %s\n", i + 1, inventory[i]);
    printf("-----------------\n");
}

//function to see if the items selected is already present in the inventory or not
int has_inventory(const char *item)
```

```c
{
    for (int i = 0; i < inventory_count; i++)
        if (strcmp(inventory[i], item) == 0)
            return 1;
    return 0;
}




// Room 1: Chamber of Light
int room1()
{
    exitroom1 = -1;
    system("cls");
    printf("\nYou have entered The Chamber of Light.\n");
    printf("\nYou step into a room bathed in radiant brilliance so pure and intense
it blurs the edges of reality. Polished mirrors line the walls, reflecting beams
from a single source: a floating crystal pulsing with inner light. The air hums
with silent tension, and every reflection feels like its watching, waiting.\n");
    printf("\nNothing casts a shadow here...  Except you.\n");
    printf("A soft chime rings out, and words appear etched in glowing script
across the stone:\n");
    printf("Truth lies only where light cannot reach. Reveal what the eye was never
meant to see.\n");
    printf("Countinue with your journey:\n");
    int choice = 0;


    while (exitroom1 == -1)
    {
        printf("\nWhich area would you like to search for the hints:\n");
        printf("1. Inspect the shelf\n");
        printf("2. Peer into the black hole\n");
        printf("3. Approach the mirror\n");
        printf("4. Exit the room\n");
        printf("\nEnter your choice:\n ");
        scanf("%d", &choice);

        switch (choice)
        {
            case 1:
            {
                system("cls");
                printf("\nYou inspect the shelf and find a paper:\n");
                printf("\nIn optics, white light through a prism splits into
distinct colors.\n");
                printf("How many primary colors appear in the visible
spectrum?\n");
                if(difficulty_mode==1)
                {
```

```c
                if (try>0)
                    {
                        char want[10];
                        printf("\nThat's why they say to keep focus in your class,
do you need a hint, enter Yes or No (Remember this will cost you your time):\n");
                        scanf("%s",want);
                        if (strcmp(want,"Yes")==0||strcmp(want,"yes")==0)
                        {
                            printf("\nHint : Here we are not talking about the
seven colors of rainbow, the answer is the primary using which other colours are
made.\n");

                                gameTimer.duration -= 30;

                        }
                        else
                        {
                            printf("\nYou have chosen not to take the hint.\n");
                        }
                    }
                }
                else if (difficulty_mode==2)
                {
                    if (try>0)
                    {
                        printf("\nYou have choosen the difficult mode , you get no
hints.\n");
                    }
                }
                else
                {
                    printf("\nHint : Here we are not talking about the seven colors
of rainbow, the answer is the primary using which other colours are made.\n");
                    printf("Hint : Think about how we taught how to make different
colours.\n");
                }

                printf("\nGive the answer for the following question(in
number):\n");
                int ans;
                scanf("%d", &ans);

                if (ans == 3)
                {
                    printf("\nCorrect! First digit of the escape code
acquired!\n");
                    achievements[4]=1;
                    code5 = 3;
                    gameTimer.duration += 30;
                    musicchange++;
                    printf("\nBonus! 30 seconds added.\n");
```

```c
                    printf("\nPress any key to Countinue...\n");
                    _getch();
                    system("cls");
                }
                else
                {
                    printf("\nWrong answer. Keep exploring.\n");
                    if(difficulty_mode==2)
                    {
                        printf("\nYou have lost 10sec due to wrong answer.\n");
                        gameTimer.duration-=10;
                    }
                    printf("\nPress any key to Countinue...\n");
                    _getch();
                    system("cls");
                    try++;
                }

                break;
            }

            case 2:
                system("cls");
                printf("\nYou peer into the black hole; a cold breeze pushes you
back. Too dangerous!\n");
                printf("\nPress any key to Countinue...\n");
                getch();
                system("cls");
                break;

            case 3:
            {
                system("cls");

                printf("\nYou touch the mirror and a riddle appears:\n");
                printf("\nI can fill a room but take up no space. What am I?\n");
                char want[10];
                if(difficulty_mode==1)
                {
                    if (try2>0)
                    {
                        printf("\nCan't solve a simple puzzle, do you need help?
But remember it will cost you your time and you have limited time. Enter Yes or
No:\n");
                        scanf("%s",want);
                        if (strcmp(want,"Yes")==0||strcmp(want,"yes")==0)
                        {
                            printf("\nHint : Keep in mind the answer is something
related to you and is something where you are present. It is not air.\n");
                            gameTimer.duration -= 30;
```

```c
                }
                else
                {
                    printf("\nYou have chosen not to take the hint.\n");
                }
            }
        }
        else if (difficulty_mode==2)
        {
            if (try>0)
            {
                printf("\nYou have choosen the difficult mode , you get no
hints.\n");
            }
        }
        else
        {
            printf("\nHint : Keep in mind the answer is something related
to you and is something where you are present. It is not air.\n");
            printf("Hint : It helps you see things in your
surroundings.\n");
        }

        printf("\nGive the answer you think is correct for the given
puzzle:\n ");
        char answer[50];
        scanf("%49s", answer);

        if (strcasecmp(answer, "light") == 0)
        {

            printf("\nThe mirror shimmers and drops a Glowing Shard!\n");
            printf("Do you want to pick up the droped Glowing Shard? Yes or
No\n");
            char store[10];
            scanf("%s",store);
            if(strcmp(store,"yes")==0 || strcmp(store,"Yes")==0)
            {
                add_to_inventory("Shard");
                printf("You have picked up the glowing shard.");
                achievements[5]=1;
                gameTimer.duration += 20;
                printf("\nBonus! 20 seconds added.\n");
                printf("\nPress any key to Countinue...\n");
        getch();
        system("cls");
            }
            else
            {
                printf("You chose to not pick the Glowing Shard.");
```

```c
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                }

            }
            else
            {
                printf("\nThe mirror cracks slightly. Wrong answer.\n");
                if(difficulty_mode==2)
                {
                    printf("You have lost 10sec due to wrong answer.\n");
                    gameTimer.duration-=10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try2++;
            }

            break;
        }

        case 4:
            exitroom1= 0;
            if (try==0&&try2==0)
            {
                achievements[0]=1;
            }
            system("cls");
            break;

        default:
            if (timerFinished==1 && gameRunning==0)
            {
                printf("Press 4 to go the restart option menu.\n");
            }
            else
            {
            printf("\nInvalid choice. Try again.\n");
            }
        }
    }

    return code5;
}


// Room 2 : The Echo Library
int room2()
```

```c
{
    exitroom2=-1;
    int choice = 0;
    char input[100];
    system("cls");
    printf("\nYou step into the Echo Library. The doors shut silently behind
you.\n");
    printf("Dust floats like memory in the air. Towering shelves stretch into
shadows above, filled with forgotten tomes.\n");
    printf("A disembodied whisper loops faintly: \"Only truth can read what was
hidden... even silence lies.\"\n");

    while (exitroom2 == -1)
    {
        printf("\nWhere would you like to explore?\n");
        printf("1. Approach the Riddle Pedestal\n");
        printf("2. Inspect the Cipher Cabinet\n");
        printf("3. Listen to the Whispering Walls\n");
        printf("4. Exit the Library\n");
        printf("\nEnter your choice:\n ");
        scanf("%d", &choice);
        getchar();
        system("cls");

        switch (choice)
         {
            case 1:
            {
                system("cls");
                printf("\nYou stand before the levitating book. A spectral quill
begins to write:\n");
                printf("\nI speak without a mouth and hear without ears. I have no
body, but I come alive with wind.\n");

                if (difficulty_mode == 1 && try7 > 0)
                {
                    char wantHint[10];
                    printf("\nDo you want a hint? This will cost you 30 seconds.
(Yes/No):\n");
                    scanf("%s", wantHint);
                    getchar();
                    if (strcasecmp(wantHint, "yes") == 0) {
                        printf("\nHint: What repeats your words in a canyon?\n");
                        gameTimer.duration -= 30;
                    }
                }
                else if (difficulty_mode == 2 )
                {
                    printf("\n No hints given since you have chosen the hard
mode.\n");
```

```c
            }
            else if(difficulty_mode==0)
            {
                printf("\nHint : What repeats your words in a canyon ?\n");
                printf("Hint : How do bats locate where they are going?\n");
            }

            printf("Your Answer: ");
            fgets(input, sizeof(input), stdin);
            input[strcspn(input, "\n")] = '\0';

            if (strcasecmp(input, "echo") == 0)
            {
                musicchange++;
                printf("\nThe book glows. The sound dome dissipates.\n");
                printf("A whisper forms into a note in mid-air:\n");
                printf("Note Acquired: 'The key is trapped in silence. Break
the code, and it shall sing.\n");
                achievements[6] = 1;
                gameTimer.duration += 30;
                printf("\nBonus! 30 seconds added.\n");
                printf("\nPress any key to Countinue...\n");
            _getch();
            system("cls");
            }
            else
            {
                printf("\nThe tome shivers.That is not the truth.\n");
                if (difficulty_mode == 2)
                {
                    printf("\nTime penalty! -10 seconds.\n");
                    gameTimer.duration -= 10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try7++;
            }
            break;
        }

        case 2:
        {
            system("cls");
            printf("\nYou inspect the locked cabinet. \n");
            printf("Born of ink but speak I may, I hold the minds of those far
away. I whisper truth when I am read, Yet stay forever, never dead. What am I?\n");

            if (difficulty_mode == 1 && try8 > 0) {
                char hintAsk[10];
```

```c
                printf("Need a hint? This costs 30 seconds. (Yes/No): ");
                scanf("%s", hintAsk);
                getchar();
                if (strcasecmp(hintAsk, "yes") == 0)
                {
                    printf("\nHint: It holds knowledge, yet it cannot
think.\n");

                    gameTimer.duration -= 30;
                }
            }
            else if (difficulty_mode == 2)
            {
                    printf("\nYou have chosen the difficult mode , you get no
hints.\n");
            }
            else if(difficulty_mode==0)
            {
                printf("\nHint: You turn its body to hear its voice.");
                printf("\nHint: It lives on shelves, speaks through pages, and
waits to be read.\n");
            }

            printf("\nDecoded Word:\n");
            fgets(input, sizeof(input), stdin);
            input[strcspn(input, "\n")] = '\0';

            if (strcasecmp(input, "books") == 0||strcasecmp(input, "book") ==
0)
            {
                printf("\nThe cabinet clicks open. Inside is an obsidian tablet
pulsing faintly.\n");
                printf("\nItem Acquired: Obsidian Tablet\n");
                add_to_inventory("Obsidian Tablet");
                gameTimer.duration += 20;
                printf("\nBonus! 20 seconds added.\n");
                code6 = 6;
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
            } else {
                printf("The cipher remains locked.\n");
                if (difficulty_mode == 2)
                {
                    printf("Time penalty! -10 seconds.\n");
                    gameTimer.duration -= 10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try8++;
```

```c
                }
                break;
        }

        case 3:
            {
            system("cls");
            printf("\nYou press your ear to the walls. The whispers shift and
resolve into a phrase:\n");
            printf("Not all echoes fade... some become warnings.\n");
            printf("\nA chill runs down your spine. Nothing more reveals
itself.\n");
            printf("\nPress any key to Countinue...\n");
            _getch();
            system("cls");
            break;
            }
        case 4:
            exitroom2 = 0;
            if (try7 == 0 && try8 == 0) {
                achievements[1] = 1;
            }
            printf("\nYou step back into the corridor. The silence
follows...\n");
            break;

        default:
            if (timerFinished && !gameRunning) {
                printf("Press 4 to return to the restart menu.\n");
            } else {
                printf("Invalid choice. Try again.\n");
            }
        }
    }

    return code6;
}


// Room 3 : The Machine Core
int room3()
{
    exitroom3 = -1;
    system("cls");
    printf("\nThe door hisses open with a blast of steam. You step into a chamber
of grinding gears, flickering lights, and pulsing conduits. Iron walls shudder with
each thrum of the colossal engine at the heart of the room a core of ancient
machinery, alive with forgotten purpose. The air smells of oil and ozone. Broken
control panels blink sporadically, as if remembering how to function. Somewhere
```

```c
above, a metallic voice loops a distorted phrase:\n");
    printf("\nCycle.... interrupted. Error: Consciousness unresolved.\n");
    printf("\nPipes snake like veins across the floor, some leaking glowing fluid,
others sparking with volatile energy. A central console stands intact its interface
locked behind a cipher pulsing in time with your heartbeat. Here, logic is law. But
something within the machine thinks... and its waiting.\n");

    int choice=0;
    while(exitroom3==-1)
    {
        printf("\nWhich area would you like to search for the hints:\n");
        printf("1. Solve The Binary Lock\n");
        printf("2. Inspect The CPU chamber\n");
        printf("3. enter the void box\n");
        printf("4. Exit the room\n");
        printf("\nEnter your choice:\n ");
        scanf("%d", &choice);
        system("cls");

        switch (choice)
        {
            case 1:
            {
                system("cls");
                printf("\nCovert The Following code From Binary To Decimal:\n");
                printf("\nThe Code:  0000 1001\n");
                if(difficulty_mode==1)
                {
                if (try3>0)
                    {
                        char want[10];
                        printf("\n Do you need a hint, enter Yes or No (Remember
this will cost you your time):\n");
                        scanf("%s",want);
                        if (strcmp(want,"Yes")==0||strcmp(want,"yes")==0)
                        {
                            printf("\nHint : Think about the approach we use to
convert Decimal to Binary.Just do the reverse in this case.\n");
                            gameTimer.duration -= 30;
                        }
                        else
                        {
                            printf("\nYou have chosen not to take the hint.\n");
                        }
                    }
                }
                else if (difficulty_mode==2)
                {
                    if (try3>0)
                    {
```

```c
                    printf("\nYou have choosen the difficult mode , you get no
hints.\n");
                }
            }
            else
            {
                printf("\nHint : Think about the approach we use to convert
Decimal to Binary.Just do the reverse in this case.\n");
                printf("Hint : Its something about adding powers of 2\n");
            }

            printf("\nGive the answer for the following question(in
number):\n");
            int ans;
            scanf("%d", &ans);

            if (ans == 9)
            {
                printf("\nCorrect! Third digit of the escape code
acquired!\n");
                achievements[7]=1;
                code7 = 9;
                gameTimer.duration += 30;
                musicchange++;
                printf("\nBonus! 30 seconds added.\n");
                printf("\nPress any key to Countinue...\n");
            _getch();
            system("cls");
            }
            else
            {
                printf("\nWrong answer. Keep exploring.\n");
                if(difficulty_mode==2)
                {
                    printf("\nYou have lost 10sec due to wrong answer.\n");
                    gameTimer.duration-=10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try3++;
            }

            break;
        }
        case 3:
        {
            system("cls");
            printf("\nYou entered  the void box; The void box is emty!!!\n");
            printf("\nPress any key to Countinue...\n");
```

```c
            _getch();
            system("cls");
            break;
        }
        case 2:
        {
            system("cls");
            printf("\nYou inspect the CPU Chamber and find a piece of paper
containing a riddle:\n");
            printf("\n I am the brain inside the box,I keep the code inside its
locks Tasks I juggle, speed I bring,Without me, the machine does nothing.\n");
            char want[10];
            if(difficulty_mode==1)
            {
                if (try4>0)
                {
                    printf("\nCan't solve a simple puzzle, do you need help?
But remember it will cost you your time and you have limited time. Enter Yes or
No:\n");
                    scanf("%s",want);
                    if (strcmp(want,"Yes")==0||strcmp(want,"yes")==0)
                    {
                        printf("\n Hint : I'm large, flat, and everything plugs
into me.\n");

                        gameTimer.duration -= 30;
                    }
                    else
                    {
                        printf("\nYou have chosen not to take the hint.\n");
                    }
                }
            }
            else if (difficulty_mode==2)
            {
                if (try4>0)
                {
                    printf("\nYou have choosen the difficult mode , you get no
hints.\n");
                }
            }
            else
            {
                printf("\n Hint : I'm large, flat, and everything plugs into
me.\n");
                printf("Hint : My name sounds like a parent who holds things
together.\n");
            }

            printf("\nGive the correct answer(in lowercase)\n ");
            char answer[50];
```

```c
            scanf("%49s", answer);

            if (strcasecmp(answer, "motherboard") == 0)
            {

                printf("\nYou have answered the riddle correctly!!\n");
                printf("Do you want to pick up the Motherboard? Yes or No\n");
                char store[10];
                scanf("%s",store);
                if(strcmp(store,"yes")==0 || strcmp(store,"Yes")==0)
                {
                    add_to_inventory("Motherboard");
                    printf("\nYou have picked up the motherboard.\n");
                    achievements[10]=1;
                    gameTimer.duration += 20;
                    printf("\nBonus! 20 seconds added.\n");
                    printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                }
                else
                {
                    printf("\nYou have chosen not to pick up the
motherboard.\n");
                }
            }
            else
            {
                printf("\n Wrong answer ! ! !\n");
                if(difficulty_mode==2)
                {
                    printf("\nYou have lost 10sec due to wrong answer.\n");
                    gameTimer.duration-=10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try4++;
            }

            break;

        }
        case 4:
        exitroom3 = 0;
        if (try3==0&&try4==0)
        {
            achievements[2]=1;
        }
        break;
```

```c
        default:
            if (timerFinished==1 && gameRunning==0)
            {
                printf("Press 4 to go the restart option menu.\n");
            }
            else
            {
            printf("\nInvalid choice. Try again.\n");
            }
        }
    }
    return code7;
}



// Room 4 : The Eye of Silence
int room4()
{
    exitroom4 = -1;
    system("cls");
    printf("\nYou cross the threshold into stillness true and absolute. No wind, no
breath, not even the sound of your own footsteps. The chamber is vast and bare, lit
by a cold, ambient glow that has no source. High above, suspended in darkness, a
single stone eye gazes down, unblinking.\n");
    printf("It does not move.\n");
    printf("It does not speak.\n");
    printf("But you feel its judgment all the same. Words, etched into the floor
beneath you, fade into view:\n");
    printf("\nThe past remembers. The Eye sees. Speak only what was earned.\n");
    printf("Scattered around the room are fragments - symbols, numbers, echoes of
choices you have made. The silence weighs heavier with each step, pressing against
your thoughts. Here, memory is not recalled - it is tested.\n");

    int choice = 0;
    int try5 = 0;
    int try6 = 0;

    while (exitroom4==-1)
    {
        printf("\nWhat would you like to do in the Eye of Silence?\n");
        printf("1. Examine the stone eye\n");
        printf("2. Search among the scattered fragments\n");
        printf("3. Study the floor inscriptions\n");
        printf("4. Exit the room\n");
        printf("\nEnter your choice:\n ");
        scanf("%d", &choice);
        system("cls");
```

```c
switch (choice)
{
    case 1:
    {
        system("cls");
        printf("\nAs you focus on the stone eye, it seems to follow your
movements. A spectral voice whispers:\n");
        printf("\nWhen one reflects on silence, what number holds the
weight of all that remains unsaid?\n");
        printf("The stone eye blinks once, awaiting your answer.\n");

        if(difficulty_mode == 1)
        {
            if (try5 > 0)
            {
                char want[10];
                printf("\nThe eye's gaze intensifies. Do you need a hint?
Enter Yes or No (Remember this will cost you your time):\n");
                scanf("%s", want);
                if (strcmp(want, "Yes") == 0 || strcmp(want, "yes") == 0)
                {
                    printf("\nHint : Think about the sense that is never
used in complete silence. How many of this sense do you have?\n");
                    gameTimer.duration -= 30;
                }
            }
        }
        else if (difficulty_mode == 2)
        {
            if (try5 > 0)
            {
                printf("\nYou have chosen the difficult mode, you get no
hints.\n");
            }
        }
        else
        {
            printf("\nHint : Think about the sense that is never used in
complete silence. How many of this sense do you have?\n");
            printf("Hint : Consider what parts of your body become useless
when there is no sound.\n");
        }

        printf("\nEnter your answer to the eye's question(The answer is in
NUMBER):\n");
        int ans;
        scanf("%d", &ans);

        if (ans == 2)
        {
```

```c
                printf("\nThe eye blinks slowly in approval. The number 2, for
the ears that hear nothing in perfect silence.\n");
                achievements[8] = 1;
                code8 = 4;
                gameTimer.duration += 30;
                musicchange++;
                printf("\nBonus! 30 seconds added.\n");
                printf("\nPress any key to Countinue...\n");
            _getch();
            system("cls");
            }
            else
            {
                printf("\nThe eye narrows in disapproval. Wrong answer.\n");
                if(difficulty_mode == 2)
                {
                    printf("\nYou have lost 10sec due to wrong answer.\n");
                    gameTimer.duration -= 10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try5++;
            }
            break;
        }

        case 2:
        {
            system("cls");
            printf("\nYou examine the scattered fragments across the room. When
you approach them, they begin to glow and form a pattern of numbers:\n");
            printf("\n8  27  64  ? \n");
            printf("\nWhat number completes this sequence?\n");

            if(difficulty_mode == 1)
            {
                if (try6 > 0)
                {
                    char want[10];
                    printf("\nThe numbers shimmer and blur. Do you need a hint?
Enter Yes or No (This will cost you time):\n");
                    scanf("%s", want);
                    if (strcmp(want, "Yes") == 0 || strcmp(want, "yes") == 0)
                    {
                        printf("\nHint : Think about perfect cubes. Each number
is a perfect cube of something...\n");
                        gameTimer.duration -= 30;
                    }
                }
```

```c
                }
                else if (difficulty_mode == 2)
                {
                    if (try6 > 0)
                    {
                        printf("\nYou have chosen the difficult mode, you get no
hints.\n");
                    }
                }
                else
                {
                    printf("\nHint : Think about perfect cubes. Each number is a
perfect cube of something...\n");
                    printf("Hint : 8 is 2^3, 27 is 3^3, 64 is 4^3... what would be
next?\n");
                }

                int ans;
                printf("\nEnter the next number in the sequence:\n");
                scanf("%d", &ans);

                if (ans == 125)
                {
                    printf("\nThe fragments pulse with bright light and arrange
themselves perfectly. 125 is indeed the next number!\n");
                    printf("(8=2^3, 27=3^3, 64=4^3, 125=5^3 - the sequence of
perfect cubes)\n");

                    if (!has_inventory("Symbol Key"))
                    {
                        printf("\nAs the fragments settle, they crystallize into a
strange key made of light - a Symbol Key!\n");
                        printf("Do you want to pick up the Symbol Key? Yes or
No\n");
                        char store[10];
                        scanf("%s", store);
                        if(strcmp(store, "yes") == 0 || strcmp(store, "Yes") == 0)
                        {
                            add_to_inventory("Symbol Key");
                            gameTimer.duration += 20;
                            printf("\nBonus! 20 seconds added.\n");
                            printf("\nPress any key to Countinue...\n");
                            _getch();
                            system("cls");
                        }
                    }
                }
                else
                {
                    printf("\nThe fragments scatter chaotically. Wrong answer.\n");
```

```c
                if(difficulty_mode == 2)
                {
                    printf("\nYou have lost 10sec due to wrong answer.\n");
                    gameTimer.duration -= 10;
                }
                printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                try6++;
            }
            break;
        }

        case 3:
        {
            system("cls");
            printf("\nYou kneel to study the inscriptions on the floor. As you
trace the words with your fingers,\n");
            printf("New text appears beneath your touch:\n");
            printf("\n\"In the language of silence, truth is revealed. What is
seen cannot be unheard.\"\n");
            printf("\nBeneath these words, four symbols are arranged in a
circle:\n");

            if (has_inventory("Symbol Key"))
            {
                printf("\nYour Symbol Key resonates with the floor
inscriptions!\n");
                printf("The symbols rearrange themselves to form a
message:\n");
                printf("\"The final number is the count of chambers where
silence truly speaks.\"\n");

                printf("\nHow many rooms in the vault embody true silence?
(Enter a number)\n");
                int ans;
                scanf("%d", &ans);

                if (ans == 1)
                {
                    printf("\nCorrect! Only this chamber - The Eye of Silence -
embodies true silence.\n");
                    printf("You have confirmed your final code digit: 4\n");
                    printf("\nPress any key to Countinue...\n");
                _getch();
                system("cls");
                }
                else
                {
                    printf("\nThe symbols scatter and fade. That is not
```

```c
correct.\n");
                    }
                }
                else
                {
                    printf("\nThe symbols seem meaningful, but you can't decipher
them. Perhaps you need a key.\n");
                    printf("\nPress any key to Countinue...\n");
                    _getch();
                    system("cls");
                }
                break;
            }

            case 4:
                exitroom4 = 0;
                if (try5 == 0 && try6 == 0 && has_inventory("Symbol Key"))
                {
                    achievements[3] = 1;
                }
                break;

            default:
                if (timerFinished == 1 && gameRunning == 0)
                {
                    printf("Press 4 to go to the restart option menu.\n");
                }
                else
                {
                    printf("\nInvalid choice. Try again.\n");
                }
        }
    }

    return code8;
}



// Main puzzle loop
void ask_puzzles(int (*has_item)(const char *),
                int (*r1)(), int (*r2)(), int (*r3)(), int (*r4)(),void
(*Achievements)(),void (*showinventory)(),void (*animatedCredits)())
{
    int choice;
    int code1 = 0, code2 = 0, code3 = 0, code4 = 0;
    char finalcode[5] = {0};
    const char *target = "3694";

    while (gameRunning && !timerFinished)
```

```c
{
    printf("\nWhich room would you like to enter?\n");
    printf("1. Chamber of Light\n");
    printf("2. Echo Library\n");
    printf("3. Machine Core\n");
    printf("4. The Eye of Silence\n");
    printf("5. Check Inventory\n");
    printf("6. Try to escape\n");
    printf("7. Pause game\n");
    printf("8. Exit Game\n");

    printf("\nEnter your Choice:\n");
    scanf("%d", &choice);

    switch (choice)
    {
        case 1:
        {
            int result = r1();
            if (result > 0)
            {
                code1 = result;
                system("cls");
                printf("\nChamber of Light solved! Code digit = %d\n",
code1);
            }
            else
            {
                system("cls");
                printf("\nLeft Chamber of Light without solving.\n");
            }
            break;
        }

        case 2:
            if (has_item("Shard") && has_item("Motherboard"))
                {
                int result = r2();
                if (result > 0)
            {
                code2= result;
                system("cls");
                printf("\nEcho Library solved! Code digit = %d\n", code1);
            }
            else
            {
                system("cls");
                printf("\nLeft Echo Library without solving.\n");
            }
```

```c
                break;
                }
            else  if(!has_item("Shard") && has_item("Motherboard"))
                {
                    printf("\nDoor sealed. You need a something from Room
1.\n");

                }
            else if(has_item("Shard") && !has_item("Motherboard"))
                {
                    printf("\nDoor sealed . You need something from Room
3.\n");

                }
            else
                {
                    printf("\nDoor sealed. You need a items from both Room
1 and 3 .\n");

                }
                break;

        case 3:
        {
            int result = r3();
            if (result > 0)
            {
                code3 = result;
                system("cls");
                printf("\n The Machine Core  solved! Code digit = %d\n",
code3);
            }
            else
            {
                system("cls");
                printf("\n Left Machine Core without solving.\n");
            }
            break;
        }

        case 4:
        {
            int result= r4();
            if (result > 0)
            {
                code4 = result;
                system("cls");
                printf("\n The Eye of Silence solved! Code digit = %d\n",
code4);
            }
            else
            {
                system("cls");
```

```c
                        printf("\n Left Eye of Silence without solving.\n");
                    }
                    break;
                }

                case 5:
                {
                    show_inventory();
                    printf("\nWould you like to clear the screen:(yes/no)\n");
                    char str[10];
                    scanf("%s",str);
                    if (strcmp(str,"yes")==0 || strcmp(str,"Yes")==0)
                    {
                        system("cls");
                    }
                    break;
                }

                case 6:
                    finalcode[0] = '0' + code1;
                    finalcode[1] = '0' + code2;
                    finalcode[2] = '0' + code3;
                    finalcode[3] = '0' + code4;
                    finalcode[4] = '\0';

                    if (strcmp(finalcode, target) == 0)
                    {
                        musicchange=10;
                        Sleep(1000);
                        PlaySound("openvault.wav", NULL, SND_FILENAME | SND_LOOP |
SND_ASYNC);
                        Sleep(5000);
                        PlaySound(NULL, 0, 0);
                        system("cls");
                        printf("\n Congratulations! You've escaped The Vaults of
Echoes! \n");
                        achievements[9]=1;
                        showinventory();
                        Achievements();
                        gameRunning = 0;
                        gameTimer.running = 0;
                        printf("\nPress any key to Countinue...\n");
                        _getch();
                        animatedCredits();
                    }
                    else
                    {
                        printf("\nWrong code (%s). Keep trying!\n", finalcode);
                    }
                    break;
```

```c
case 7:
{
    if (!game_paused)
    {
        game_paused = 1;
        ResetEvent(pause_event);
        system("cls");
        printf("\n====================================\n");
        printf("              GAME PAUSED\n");
        printf("====================================\n");
        printf("Time remaining: %02d:%02d\n",
gameTimer.duration / 60, gameTimer.duration % 60);
        printf("\nOptions:\n");
        printf("1. Resume game\n");
        printf("2. View inventory\n");


        int pause_choice;
        printf("Enter your choice: ");
        scanf("%d", &pause_choice);

        switch (pause_choice)
        {
        case 1:
        {
            printf("Resuming game...\n");
            printf("\nPress any key to Countinue...\n");
            _getch();
            system("cls");
            break;
        }
        case 2:
            show_inventory();
            printf("\nPress any key to Countinue...\n");
            _getch();

            break;
        default:
            printf("Invalid choice. Resuming game...\n");
        }

        game_paused = 0;
        SetEvent(pause_event);
    }
    else
    {
        printf("Game is already paused!\n");
    }
```

```c
                    break;
                }

                case 8:
                    gameRunning = 0;
                    exitgame=1;
                    gameTimer.running = 0;
                    break;

                default:
                if (timerFinished==1 && gameRunning==0)
                {
                    printf("\n");
                }
                else
                {
                    printf("\nInvalid choice. Try again.\n");
                }
            }

        }
}


void playMusic(const char *filename)
{
    PlaySound(filename, NULL, SND_FILENAME | SND_ASYNC | SND_LOOP);
}

void stopMusic()
{
    PlaySound(NULL, NULL, 0);
}

void animatedCredits()
{
    const char *credits[] = {
        "                                    ====================================",
        "                                         THE  VAULT  OF  ECHOES         ",
        "                                    ====================================",
        "                                                                        ",
        "                                      A Text-Based Puzzle Adventure Game ",
        "                                     An fisrt of a kind Online EscapeRoom",
        "                                                                        ",
        "                                                                        ",
        "                                    Designed and Programmed By:",
        "                                    >> Arin Singh",
        "                                    >> Apeksha Ashok",
        "                                    >> Ashritha Hegde",
        "                                    >> Aryan Nangarath",
```

```c
                    "
",
                    "
",
                    "                                        Background Music: ",
                    "                                        BGM Musics",
                    "                                        Video Game Musics",
                    "                                        Bensounds",
                    "
",
                    "                                        Special Thanks TO:",
                    "                                        ChatGPT ",
                    "                                        DeepSeek",
                    "
",
                    "                                            Thank you for playing!",
                    "
",
                    "                                                ---END CREDITS---    ",
                    "",
                    NULL
    };

    playMusic("bright.wav");

    int totalLines = 0;
    while (credits[totalLines])
    {
    totalLines++;
    }
    for (int offset = 0; offset < totalLines + 25; offset++) {
        system("cls");
        for (int i = 0; i < 25; i++) {
            int lineIndex = offset - (25 - i);
            if (lineIndex >= 0 && lineIndex < totalLines) {
                printf("%*s%s\n", 20, "", credits[lineIndex]);
            } else {
                printf("\n");
            }
        }
        Sleep(450);
    }

    stopMusic();
    printf("\n\nPress any key to exit credits...");
    _getch();
}

// For the blinding light and crash sound
```

```c
HANDLE hConsole;
void simulate_blinding_light() {
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    CONSOLE_SCREEN_BUFFER_INFO consoleInfo;
    WORD originalAttributes;
    GetConsoleScreenBufferInfo(hConsole, &consoleInfo);
    originalAttributes = consoleInfo.wAttributes;
    PlaySound("crash.wav", NULL, SND_FILENAME | SND_ASYNC | SND_LOOP);
    Sleep(5000);
    PlaySound(NULL, NULL, 0);
    SetConsoleTextAttribute(hConsole, BACKGROUND_RED | BACKGROUND_GREEN |
BACKGROUND_BLUE);
    system("cls");
    Sleep(1500);

    SetConsoleTextAttribute(hConsole, originalAttributes);
    system("cls");
    Sleep(500);
}


//Main Function
int main()
{

    hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    simulate_blinding_light();

printf("=========================================================================
======================================================\n");
    printf("                                                        THE VAULT
OF ECHOES                                                \n");

printf("=========================================================================
======================================================\n");
    printf("\nA deafening crash. A blinding white light. Then nothing.\n");
    printf("\nYou awaken alone in a vast, stone-carved labyrinth where every
footstep echoes like a heartbeat in the dark. The air is thick with silence until a
voice, cold and disembodied, slithers through the gloom:\n");
    printf("\n""Your memories lie shattered across four sealed chambers, deep
beneath the earth. Solve their riddles. Reclaim the fragments of your soul. Unlock
the final code... before the Vaults gates close forever.""\n");
    printf("\nWill you escape or will the Vault of Echoes become your tomb?\n");
    Sleep(10000);
    printf("\nChoose your Difficulty Mode:\n");
    printf("0 - Easy   (More Time, Hints are Free and No Penalty)\n");
    printf("1 - Normal (Standard Game and 5 min Time limit)\n");
    printf("2 - Hard   (Less Time, No Hints, High Penalties)\n");

    do
```

```c
{
    printf("Enter your choice (0/1/2): ");
    scanf("%d", &difficulty_mode);
}
while (difficulty_mode < 0 || difficulty_mode > 2);
Sleep(500);
system("cls");
switch (difficulty_mode)
{
    case 0:
        gameTimer.duration = 7 * 60;
        break;
    case 1:
        gameTimer.duration = 5 * 60;
        break;
    case 2:
        gameTimer.duration = 3 * 60 ;
        break;
}

while (user_input())
{
    gameRunning     = 1;
    timerFinished   = 0;
    inventory_count = 0;
    musicchange     = 0;
    exitroom1 = exitroom2 = exitroom3 = exitroom4 = 0;
    gameTimer.running  = 1;

    switch (difficulty_mode)
    {
        case 0:
            gameTimer.duration = 7 * 60;
            break;
        case 1:
            gameTimer.duration = 5 * 60;
            break;
        case 2:
            gameTimer.duration = 3 * 60;
            break;
    }


    pause_event = CreateEvent(NULL, TRUE, TRUE, NULL);
    if (pause_event == NULL)
    {
        printf("Error creating pause event! Error: %d\n", GetLastError());
        return 1;
    }
```

```c
        HANDLE hTimer = CreateThread(NULL, 0, TimerThread, &gameTimer, 0, NULL);
        HANDLE hMusic = CreateThread(NULL, 0, MusicThread, NULL, 0, NULL);

        ask_puzzles(has_inventory, room1, room2, room3, room4,
Achievements,show_inventory,animatedCredits);

        gameRunning = 0;
        WaitForSingleObject(hTimer, INFINITE);
        CloseHandle(hTimer);

        TerminateThread(hMusic, 0);
        CloseHandle(hMusic);

        if (pause_event)
        CloseHandle(pause_event);

        if (gameTimer.duration == 0)
        {
            printf("\nTime's up! The vault claims another victim. \n");
            printf("\n      --- Timer Ended and Game Over ---     \n");
        }
        else if (gameTimer.duration > 1 && exitgame == 1  )
        {
            printf("\nYou gave up. The vault claims another victim.\n");
            printf("\n               --- Game Over ---               \n");
        }
        else
        {
            printf("\nYou have achieved the ending of the THE VAULTS OF ECHOES.\n
");
            printf("\n                   --- Game Ended ---
\n");

        }
    }
    printf("\n!!!Goodbye!!!\n");
    return 0;
}
```