**实验十**

1. **创建和编写文件alphabet\_timer.c**

**文本

AI 生成的内容可能不正确。**

源代码

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <signal.h>

#include <sys/time.h>

static int count = 0; // 记录触发次数

// SIGALRM信号处理函数

void alarm\_handler(int sig) {

    count++; // 每次信号触发时增加计数

}

int main() {

    struct sigaction sa;

    struct itimerval timer;

    // 设置信号处理函数

    sa.sa\_handler = alarm\_handler;

    sa.sa\_flags = 0;

    sigemptyset(&sa.sa\_mask);

    if (sigaction(SIGALRM, &sa, NULL) == -1) {

        perror("sigaction");

        exit(1);

    }

    // 设置定时器：初始延迟1秒，之后每隔1秒触发一次

    timer.it\_value.tv\_sec = 1;

    timer.it\_value.tv\_usec = 0;

    timer.it\_interval.tv\_sec = 1;

    timer.it\_interval.tv\_usec = 0;

    if (setitimer(ITIMER\_REAL, &timer, NULL) == -1) {

        perror("setitimer");

        exit(1);

    }

    // 主循环

    while (1) {

        pause(); // 等待信号

        // 输出当前字母（A-Z循环）

        printf("%c", 'A' + (count % 26));

        fflush(stdout); // 立即刷新输出缓冲区

    }

    return 0;

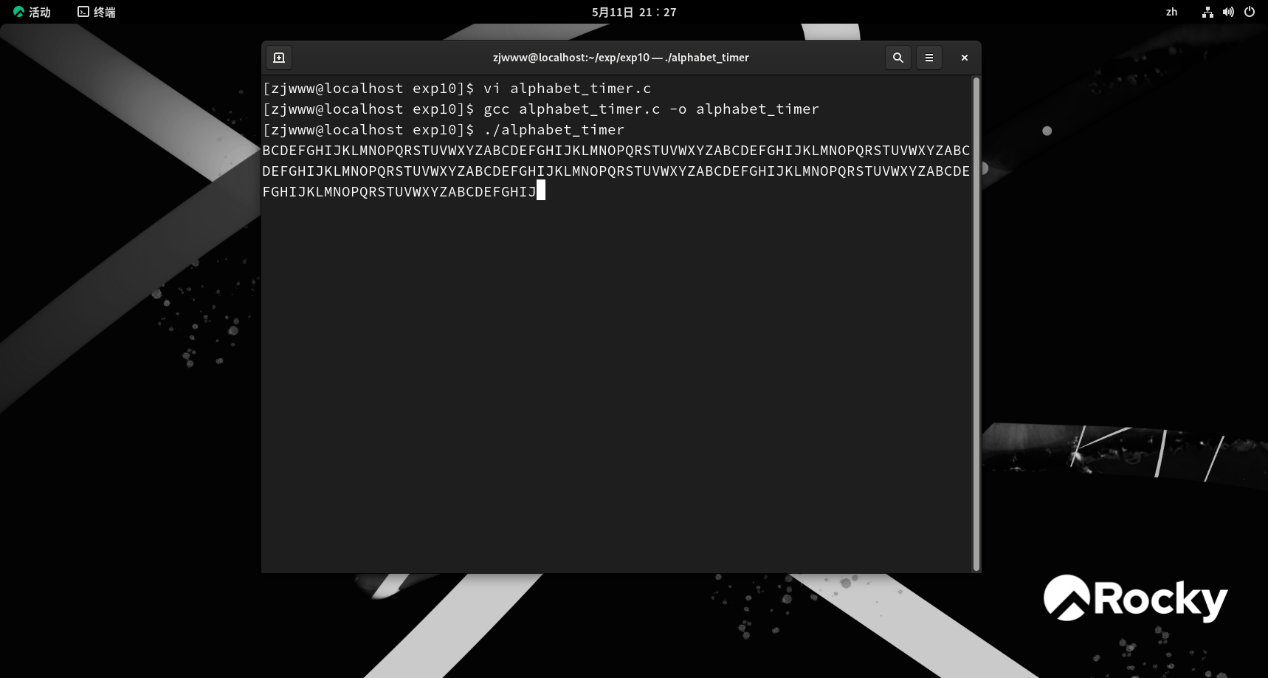
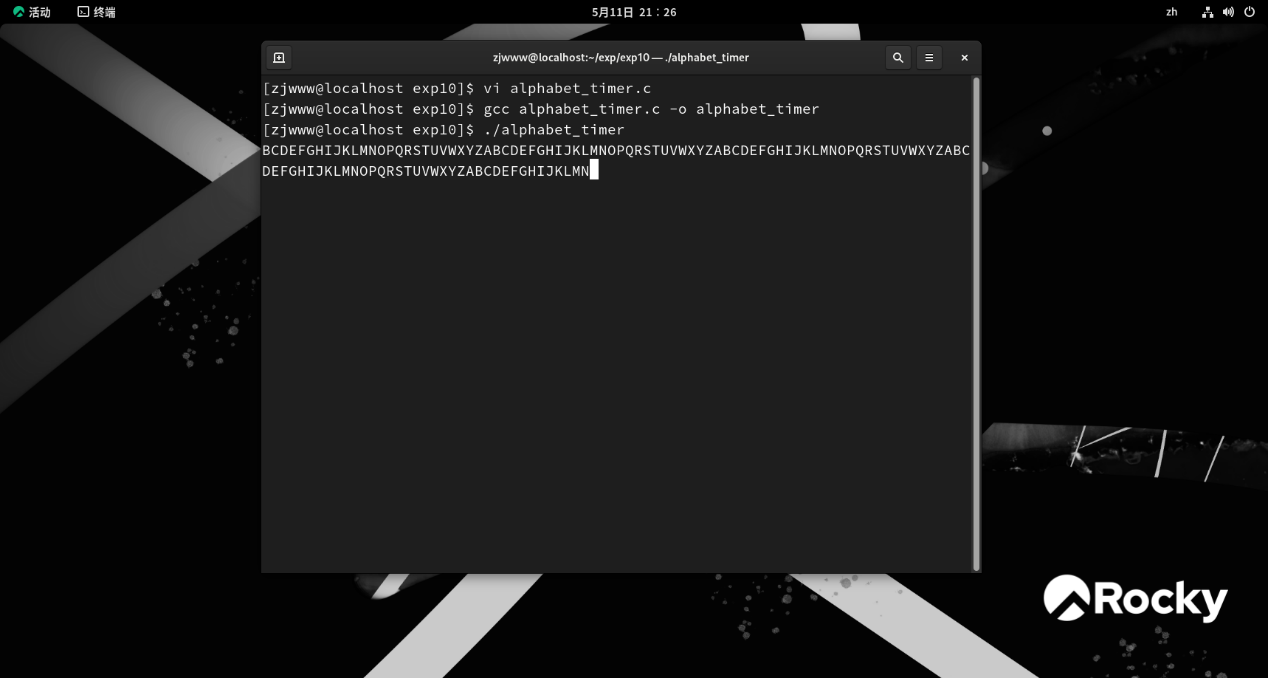
}

1. **编译运行**

**文本

AI 生成的内容可能不正确。**

**文本

AI 生成的内容可能不正确。**

**文本

AI 生成的内容可能不正确。文本

AI 生成的内容可能不正确。**