**BT 7.2**

Nguyễn Hải Dương – 20194530

1. **Bài tập 1:**



* Code:

#include <stdlib.h>

#include <stdio.h>

#include <sys/time.h>

#include <time.h>

#include <unistd.h>

float time\_diff(struct timeval \*start, struct timeval \*end)

{

    return (end->tv\_sec - start->tv\_sec) + 1e-6\*(end->tv\_usec - start->tv\_usec);

}

int main() {

    struct timeval start;

    struct timeval end;

    unsigned int seconds = 10;

    unsigned int usecs = 5;

// --------------------------------

    gettimeofday(&start, NULL);

    printf("start: %.8ld:%.8ld\n", start.tv\_sec, start.tv\_usec);

    sleep(seconds);

    gettimeofday(&end, NULL);

    printf("end: %.8ld:%.8ld\n", end.tv\_sec, end.tv\_usec);

    printf("sleep(%d)  time spent: %0.8f sec\n", seconds, time\_diff(&start, &end));

// --------------------------------

    gettimeofday(&start, NULL);

    printf("start: %.8ld:%.8ld\n", start.tv\_sec, start.tv\_usec);

    usleep(usecs);

    gettimeofday(&end, NULL);

    printf("end: %.8ld:%.8ld\n", end.tv\_sec, end.tv\_usec);

    printf("unsleep(%d) time spent: %0.8f sec\n", usecs, time\_diff(&start, &end));

// --------------------------------

    struct timespec remaining, request = { 5, 10 };

    gettimeofday(&start, NULL);

    printf("start: %.8ld:%.8ld\n", start.tv\_sec, start.tv\_usec);

    int response = nanosleep(&request, &remaining);

    gettimeofday(&end, NULL);

    printf("end: %.8ld:%.8ld\n", end.tv\_sec, end.tv\_usec);

    if (response == 0) {

        printf("Nap was Successful.\n");

    }

    else {

        printf("Nap was Interrupted.\n");

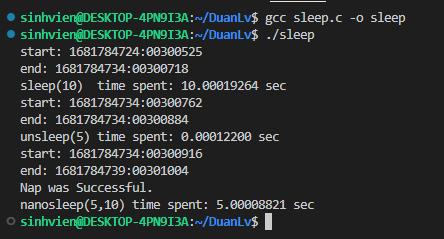
    }

    printf("nanosleep(5,10) time spent: %0.8lf sec\n", time\_diff(&start, &end));

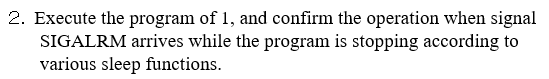
    exit(EXIT\_SUCCESS);

}

* Kết quả thực thi chương trình:



1. **Bài tập 2:**



* Code:

#include <stdlib.h>

#include <stdio.h>

#include <sys/time.h>

#include <time.h>

#include <unistd.h>

#include <signal.h>

float time\_diff(struct timeval \*start, struct timeval \*end)

{

    return (end->tv\_sec - start->tv\_sec) + 1e-6\*(end->tv\_usec - start->tv\_usec);

}

void call\_alarm(int sig)

{

    printf("ALARM CALLED\n");

}

int main() {

    struct timeval start;

    struct timeval end;

    unsigned int seconds = 10;

    unsigned int usecs = 5;

    signal(SIGALRM, call\_alarm);

// --------------------------------

    gettimeofday(&start, NULL);

    printf("start: %.8ld:%.8ld\n", start.tv\_sec, start.tv\_usec);

    sleep(seconds);

    alarm(1);

    gettimeofday(&end, NULL);

    printf("end: %.8ld:%.8ld\n", end.tv\_sec, end.tv\_usec);

    printf("sleep(%d)  time spent: %0.8f sec\n", seconds, time\_diff(&start, &end));

// --------------------------------

    signal(SIGALRM, call\_alarm);

    gettimeofday(&start, NULL);

    printf("start: %.8ld:%.8ld\n", start.tv\_sec, start.tv\_usec);

    usleep(usecs);

    alarm(1);

    gettimeofday(&end, NULL);

    printf("end: %.8ld:%.8ld\n", end.tv\_sec, end.tv\_usec);

    printf("unsleep(%d) time spent: %0.8f sec\n", usecs, time\_diff(&start, &end));

// --------------------------------

    signal(SIGALRM, call\_alarm);

    struct timespec remaining, request = { 5, 10 };

    gettimeofday(&start, NULL);

    printf("start: %.8ld:%.8ld\n", start.tv\_sec, start.tv\_usec);

    int response = nanosleep(&request, &remaining);

    alarm(1);

    gettimeofday(&end, NULL);

    printf("end: %.8ld:%.8ld\n", end.tv\_sec, end.tv\_usec);

    if (response == 0) {

        printf("Nap was Successful.\n");

    }

    else {

        printf("Nap was Interrupted.\n");

    }

    printf("nanosleep(5,10) time spent: %0.8lf sec\n", time\_diff(&start, &end));

    exit(EXIT\_SUCCESS);

}

* Thực thi chương trình để xác nhận tín hiện SIGALRM

