

Week10 Assignment

班级：212112 学号：21373339 姓名：周星达

阅读教材第六章并查阅网络资料，回答以下问题。

1. 编写一个程序，实现这样的功能：搜索2~65535之间所有的素数并保存到数组中，用户输入^C信号时，程序打印出最近找到的素数。

```
#include <unistd.h>
#include <signal.h>
#include <stdio.h>
#include <sys/types.h>
int i = 2, prime[35000], j = 0;
void sigHandler(int signalNum)
{
    printf("The prime number:%d\n", prime[j - 1]);
}
int isPrime(int n)
{
    if (n != 2 && n % 2 == 0)
        return 0;
    for (int i = 2; i * i <= n; i = i + 2)
        if (n % i == 0)
            return 0;
    return 1;
}
int main()
{
    signal(SIGINT, sigHandler);
    while (i <= 65535)
    {
        if (isPrime(i))
            prime[j++] = i;
        i++;
        usleep(1000);
    }
    return 0;
}
```

2. 简述什么是可靠信号和不可靠信号，并实验SIGINT信号是可靠的还是不可靠的。

不可靠信号：信号可能会丢失，一旦信号丢失了，进程并不能知道信号丢失 可靠信号：也是阻塞信号，当发送了一个阻塞信号，并且该信号的动作时系统默认动作或捕捉该信号，如果信号从发出以后会一直保持未决的状态，直到该进程对此信号解除了阻塞，或将对此信号的动作更改为忽略

```
#include <unistd.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <signal.h>
#include <errno.h>
#include <fcntl.h>

#define MSG "Catch signal SIGINT processing \n"
#define MSG_END "Finished process SIGINT return \n"

static void sig_handler (int signum )
{
    /*
    在信号处理程序中，尽量不要调用与标准IO相关的和不可重入的函数。
    STDIN_FILENO：接收键盘的输入
    STDOUT_FILENO：向屏幕输出
    */
    write ( STDOUT_FILENO , MSG , strlen (MSG) ) ;
    time_fun();
    write ( STDOUT_FILENO , MSG_END , strlen (MSG_END) ) ;
}

void time_fun()
{
    long long s = 0 ;
    long long i ;
    for ( i= 0 ; i < 500000000L ; i++ )
    {
        s += i ;
    }
}

int main(int argc, char **argv)
{
    // 注册信号处理函数
    if ( SIG_ERR == signal ( SIGINT , sig_handler ) )
    {
        fprintf (stderr , "signal error " ) , perror ("" ) ;
        exit (1) ;
    }

    // 让主程序不退出，挂起，等待信号产生
    while (1)
    {
        pause ( ) ; //使调用进程在接到一信号前挂起。
    }
    return 0 ;
}
```

3. 在执行 `ping http://www.people.com.cn` 时，假设该网站是可 `ping` 通的，但是在输入 `^\` 时，`ping` 命令并没有结束而是显示 `ping` 的成功率，但是输入 `^C` 时，`ping` 程序却被退出，请解释发生这一现象的原因。

`ping` 命令在程序超时或当接收到 `SIGINT` 信号时结束，键入 `^C` 发送了中断信号。`^\` 返回 `SIGQUIT` 信号 `ping` 命令会继续发送回送信号请求直到接收到中断信号

4. 简述什么是不可重入函数，为什么信号处理函数中要尽量避免包含不可重入函数？

如果信号处理函数中使用了不可重入函数，那么信号处理函数可能会修改原来进程中不应该被修改的数据，这样进程从信号处理函数中返回接着执行时，可能会出现不可预料的后果。不可重入函数在信号处理函数中被视为不安全函数。

5. 编写一个程序，实现这样的功能：程序每隔1秒就给自身发送一个信号，程序接收到该信号后，打印出当前的时间。

提示:

- 发送的信号可以是任何能实现功能的信号。
- 打印时间的格式不做限制，任何形式都是正确的。

```
#include <time.h>
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <signal.h>
#include <stdlib.h>

int n = 0;
static void sigHandler(int signum)
{
    time_t t = time(NULL);
    n++;
    printf("%d %s", n, ctime(&t));
}

int main()
{
    signal(SIGINT, sigHandler);
    pid_t pid = getpid();
    while (1)
    {
        kill(pid, SIGINT);
        sleep(1);
    }
    return 0;
}
```

6.编写程序实现如下功能：程序 **A.c** 按照用户输入的参数定时向程序 **B.c** 发送信号，**B.c** 程序接收到该信号后，打印输出一条消息。

运行过程如下：

```
./B value& # 此时，输出进程 B 的 PID 号，value 表示要输出的参数。
./A processBPID timerVal # 第一个参数表示进程 B 的 PID，第二个参数为定时时间。
```

```
#include <stdio.h>
#include <unistd.h>
#include <signal.h>
#include <sys/types.h>
#include <sys/time.h>
#include <math.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{
    int pid = atoi(argv[1]);
    int time = atoi(argv[2]);
    printf("A: send after %d seconds\n...\n", time);
    sleep(time);
    printf("A: send a signal to B\n");
    kill(pid, SIGINT);
    return 0;
}
```

```
#include <unistd.h>
#include <stdio.h>
#include <sys/types.h>
#include <signal.h>
#include <stdlib.h>
char *st[10];
int s;
void sigHandler(int signalNum)
{
    printf("B: ");
    for (int i = 1; i < s; i++)
        printf("%s ", st[i]);
    printf("\n");
}
int main(int argc, char *argv[])
{
    printf("The pid of B is %d\n", getpid());
    s = argc;
    for (int i = 1; i <= argc; i++)
        for (int i = 1; i <= argc; i++)
            st[i] = argv[i];
    signal(SIGINT, sigHandler);
}
```

```
while (1)
    ;
}
```