

Operációs Rendszerek BSc

8.Gyak

2021.03.31

Készítette:

Balogh Sándor

Mérnökinformatikus

GVVASJ

1.Feladat_Alarm

```
#include <unistd.h>
```

```
#include <signal.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#define SECOND 1
```

```
void semmi();
```

```
void do_int();
```

```
int main(){
```

```
    int i;
```

```
    unsigned sec = 1;
```

```
    signal(SIGINT, int_erkezik);
```

```
    for(i = 1; i < 8; i++){
```

```
        alarm(sec);
```

```
        signal(SIGALRM, semmi);
```

```
        printf("%d\n", i);
```

```
        pause();
```

```
    }
```

```
    return 0;
```

```
}
```

```
void semmi(){
```

```
;
```

```
}
```

```
void int_erkezik(){
```

```
    printf("int megerkezett.\n");
```

```
    signal(SIGINT, SIG_IGN);
```

```
}
```

2.Alarmra_Var

```
#include <unistd.h>
```

```
#include <signal.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
void var();
```

```
int main(){
```

```
    unsigned sec = 5;
```

```
    signal(SIGALRM, var);
```

```
    alarm(sec);
```

```
    printf("%d varok.\n",1);
```

```
    pause();
```

```
    printf("Megerkezett az alarm.\n");
```

```
    return 0;
```

```
}
```

```
void var(){
```

```
;
```

```
}
```

3.Alarm_Ado

```
#include <sys/types.h>
```

```
#include <signal.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main(int argc, char **argv){
```

```
    int pid;
```

```
    if(argc < 1){
```

```
        perror("Hiba!\n");
```

```
        exit(1);
```

```
    }
```

```
    pid = atoi(argv[1]);
```

```
    kill(pid, SIGALRM);
```

```
    return 0;
```

```
}
```

4.(TobbSign)

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <unistd.h>
```

```
#include <signal.h>
```

```
void megszakitak(int jel);
```

```
void kilepes(int jel);
```

```
unsigned int megszakitakok = 0;
```

```
int main(void)
```

```
{
```

```
    if (signal(SIGINT, megszakitak) == SIG_ERR)
```

```
    {
```

```
        printf("Nem sikerult\n");
```

```
        return 0;
```

```
    }
```

```
    if (signal(SIGQUIT, kilepes) == SIG_ERR)
```

```
    {
```

```
        printf("Nem sikerult!\n");
```

```
        return 0;
```

```
    }
```

```
    while(megszakitakok < 2)
```

```
    {
```

```
        printf("Varakozas a jelre...\n");
```

```
        sleep(1);
```

```
}
```

```
printf("Megvan a masodik SIGINT jel!");
```

```
return 0;
```

```
}
```

```
void megszakitas(int jel)
```

```
{
```

```
    printf("SIGINT jel: %d\n", jel);
```

```
    megszakitasok++;
```

```
}
```

```
void kilepes(int jel)
```

```
{
```

```
    printf("SIGQUIT jel: %d\n", jel);
```

```
}
```