

Владимир Исмагилов

# Месть программиста

```
(SPR_PUNG,2,4,(NULL),S_PUNCH5,0,0); // S_PUNCH4
(SPR_PUNG,1,5,(A_ReFire),S_PUNCH,0,0); // S_PUNCH5 // do things to change the game state
(SPR_PISG,0,1,(A_WeaponReady),S_PISTOL,0,0); // S_PISTOL while (gameaction != ga_nothing)
(SPR_PISG,0,1,(A_Lower),S_PISTOLDOWN,0,0); // S_PISTOLDOWN
(SPR_PISG,0,1,(A_Raise),S_PISTOLUP,0,0); // S_PISTOLUP
(SPR_PISG,0,4,(NULL),S_PISTOL2,0,0); // S_PISTOL1
(SPR_PISG,1,6,(A_FirePistol),S_PISTOL3,0,0); // S_PISTOL2
(SPR_PISG,2,4,(NULL),S_PISTOL4,0,0); // S_PISTOL3
(SPR_PISG,1,5,(A_ReFire),S_PISTOL,0,0); // S_PISTOL4
(SPR_PISG,3,2,7,(A_Light1),S_LIGHTDONE,0,0); // S_PISTOLFLASH
(SPR_SHTG,0,1,(A_WeaponReady),S_SGUN,0,0); // S_SGUN
(SPR_SHTG,0,1,(A_Lower),S_SGUNDOWN,0,0); // S_SGUNDOWN
(SPR_SHTG,0,1,(A_Raise),S_SGUNUP,0,0); // S_SGUNUP
(SPR_SHTG,0,3,(NULL),S_SGUN2,0,0); // S_SGUN1
(SPR_SHTG,0,7,(A_FireShotgun),S_SGUN3,0,0); // S_SGUN2
(SPR_SHTG,1,5,(NULL),S_SGUN4,0,0); // S_SGUN3
(SPR_SHTG,2,5,(NULL),S_SGUN5,0,0); // S_SGUN4
(SPR_SHTG,3,4,(NULL),S_SGUN6,0,0); // S_SGUN5
(SPR_SHTG,2,5,(NULL),S_SGUN7,0,0); // S_SGUN6
(SPR_SHTG,1,5,(NULL),S_SGUNB,0,0); // S_SGUN7
```

switch (gameaction)

```
case ga_leadlevel:
    G_DeLoadLevel ();
    break;
case ga_newgame:
    G_DeNewGame ();
    break;
case ga_loadgame:
    G_DeLoadGame ();
    break;
case ga_savegame:
    G_DeSaveGame ();
    break;
case ga_playdemo:
    G_DePlayDemo ();
```

```
// ENEMY THINKING
// Enemies are always spawned
// with targetplayer = -1, then
// Most numbers are spawned
// but some can be made pre
//
// Called by P_NoiseAlert,
// Recursively traverses adjacent
// sound blocking lines out of
//
mob_t* soundtarget;
void
P_RecursiveSound
( sector_t* sec,
  int soundblock
```

```
int i;
line_t* check;
sector_t* other;

// wake up all monsters in
if (sec->validcount == validcount)
    sec->soundtraverse = soundtarget;

return; //

sec->validcount = validcount;
sec->soundtraverse = sec;
sec->soundtarget = soundtarget;
```

```
for (i=0; i<sec->linecount; i++)
{
    check = sec->lines[i];
    if (! (check->flags & SFL_BLOCKMONSTERS))
        continue;

    P_LineOpening (check->target);

    if (openrange <= 0)
        continue; //

    cmd = &players[i].cmd;
    if (! (sideset check->sector) ||
        other = sideset check->sector)
        other = sideset check->sector;

    if (check->flags & SFL_BLOCKMONSTERS)
    {
        if (soundblock)
            P_RecursiveSound (other, sec);
        else
            P_RecursiveSound (other, sec);
    }
}
```

```
(SPR_CHGG,1,4,(A_FireCGun),S_CHAIN3,0,0); // S_CHAIN2
(SPR_CHGG,1,0,(A_ReFire),S_CHAIN,0,0); // S_CHAIN3
(SPR_CHGF,3,2,7,5,(A_Light1),S_LIGHTDONE,0,0); // S_CHAINFLASH1
(SPR_CHGF,3,2,7,5,(A_Light2),S_LIGHTDONE,0,0); // S_CHAINFLASH2
(SPR_MISG,0,1,(A_WeaponReady),S_MISSILE,0,0); // S_MISSILE
(SPR_MISG,0,1,(A_Lower),S_MISSILEDOWN,0,0); // S_MISSILEDOWN
(SPR_MISG,0,1,(A_Raise),S_MISSILEUP,0,0); // S_MISSILEUP
(SPR_MISG,1,0,(A_GunFlash),S_MISSILE2,0,0); // S_MISSILE1
(SPR_MISG,1,1,2,(A_FireMissile),S_MISSILE3,0,0); // S_MISSILE2
(SPR_MISG,1,0,(A_ReFire),S_MISSILE,0,0); // S_MISSILE3
(SPR_MISF,3,2,7,8,3,(A_Light1),S_MISSILEFLASH2,0,0); // S_MISSILEFLASH1
(SPR_MISF,3,2,7,8,4,(NULL),S_MISSILEFLASH3,0,0); // S_MISSILEFLASH2
(SPR_MISF,3,2,7,7,4,(A_Light2),S_MISSILEFLASH4,0,0); // S_MISSILEFLASH3
(SPR_MISF,3,2,7,7,4,(A_Light2),S_LIGHTDONE,0,0); // S_MISSILEFLASH4
(SPR_SAWG,2,4,(A_WeaponReady),S_SAWB,0,0); // S_SAW
(SPR_SAWG,3,4,(A_WeaponReady),S_SAWD,0,0); // S_SAWB
(SPR_SAWG,2,1,(A_Lower),S_SAWDOWN,0,0); // S_SAWDOWN
(SPR_SAWG,2,1,(A_Raise),S_SAWUP,0,0); // S_SAWUP
(SPR_SAWG,0,4,(A_Saw),S_SAW2,0,0); // S_SAW1
(SPR_SAWG,1,4,(A_Saw),S_SAW3,0,0); // S_SAW2
(SPR_SAWG,1,0,(A_ReFire),S_SAW,0,0); // S_SAW3
(SPR_PLSG,0,1,(A_WeaponReady),S_PLASMA,0,0); // S_PLASMA
(SPR_PLSG,0,1,(A_Lower),S_PLASMA2,0,0); // S_PLASMA2
(SPR_PLSG,0,1,(A_Raise),S_PLASMAUP,0,0); // S_PLASMAUP
(SPR_PLSG,0,3,(A_FirePlasma),S_PLASMA2,0,0); // S_PLASMA1
(SPR_PLSG,1,2,0,(A_ReFire),S_PLASMA,0,0); // S_PLASMA2
(SPR_PLSF,3,2,7,8,4,(A_Light1),S_LIGHTDONE,0,0); // S_PLASMAFLASH1
(SPR_PLSF,3,2,7,8,4,(A_Light1),S_LIGHTDONE,0,0); // S_PLASMAFLASH2
(SPR_BFGG,0,1,(A_WeaponReady),S_BFG,0,0); // S_BFG
(SPR_BFGG,0,1,(A_Lower),S_BFGDOWN,0,0); // S_BFGDOWN
(SPR_BFGG,0,1,(A_Raise),S_BFGUP,0,0); // S_BFGUP
(SPR_BFGG,0,2,0,(A_BFGSound),S_BFG2,0,0); // S_BFG1
(SPR_BFGG,1,1,0,(A_GunFlash),S_BFG3,0,0); // S_BFG2
(SPR_BFGG,1,1,0,(A_FireBFG),S_BFG4,0,0); // S_BFG3
(SPR_BFGG,1,2,0,(A_ReFire),S_BFG,0,0); // S_BFG4
(SPR_BFGF,3,2,7,8,11,(A_Light1),S_BFGFLASH2,0,0); // S_BFGFLASH1
(SPR_BFGF,3,2,7,8,9,6,(A_Light2),S_LIGHTDONE,0,0); // S_BFGFLASH2
(SPR_BLOOD,2,8,(NULL),S_BLOOD2,0,0); // S_BLOOD1
(SPR_BLOOD,1,8,(NULL),S_BLOOD3,0,0); // S_BLOOD2
(SPR_BLOOD,0,8,(NULL),S_NULL,0,0); // S_BLOOD3
(SPR_PUFF,3,2,7,8,4,(NULL),S_PUFF2,0,0); // S_PUFF1
(SPR_PUFF,1,4,(NULL),S_PUFF3,0,0); // S_PUFF2
(SPR_PUFF,2,4,(NULL),S_PUFF4,0,0); // S_PUFF3
(SPR_PUFF,3,4,(NULL),S_NULL,0,0); // S_PUFF4
```

```
if (playeringame[i])
{
    cmd = &players[i].cmd;
    memcpy (cmd, &netcmds[i].cmd, sizeof(ticcmd_t));

    if (demoplayback)
        G_ReadDemoTiccmd (cmd);
    if (demorecording)
        G_WriteDemoTiccmd (cmd);

    // check for turbo cheats
    if (cmd->forwardmove > TURBOTHRESHOLD
        && !(gametic&31) && (gametic>>5)&3) == i)
    {
        static char turbomessage[90];
        extern char *player_names[4];
        sprintf (turbomessage, "%s is turbo!", player_names[i]);
        players[consoleplayer].message = turbomessage;
    }

    if (netgame && !netdemo && !(gametic%ticdup))
    {
        if (gametic > BACKUPTICS
            && consistency[fillbuf] != cmd->consistency)
        {
            P_NoiseAlert (cmd->target, cmd->emitter);
            _Error ("consistency failure (%i should be %i)", mob_t* target,
                cmd->consistency, consistency[fillbuf]);
        }
    }

    if (players[i].mo)
        consistency[fillbuf] = players[i].mo->x;
    else
        consistency[fillbuf] = rindex;
}

// P_NoiseAlert,
// If a monster yells at a player
// it will alert other monsters
void
P_NoiseAlert
( mob_t* target,
  mob_t* emitter )
{
    soundtarget = target;
    validcount++;
    P_RecursiveSound (emitter, sec);
}

// Range
boolean P_ChedMeleeRange
( mob_t* p;
  fixed_t dist;
```

```
if (playeringame[i])
{
    cmd = &players[i].cmd;
    memcpy (cmd, &netcmds[i].cmd, sizeof(ticcmd_t));

    if (demoplayback)
        G_ReadDemoTiccmd (cmd);
    if (demorecording)
        G_WriteDemoTiccmd (cmd);

    // check for turbo cheats
    if (cmd->forwardmove > TURBOTHRESHOLD
        && !(gametic&31) && (gametic>>5)&3) == i)
    {
        static char turbomessage[90];
        extern char *player_names[4];
        sprintf (turbomessage, "%s is turbo!", player_names[i]);
        players[consoleplayer].message = turbomessage;
    }

    if (netgame && !netdemo && !(gametic%ticdup))
    {
        if (gametic > BACKUPTICS
            && consistency[fillbuf] != cmd->consistency)
        {
            P_NoiseAlert (cmd->target, cmd->emitter);
            _Error ("consistency failure (%i should be %i)", mob_t* target,
                cmd->consistency, consistency[fillbuf]);
        }
    }

    if (players[i].mo)
        consistency[fillbuf] = players[i].mo->x;
    else
        consistency[fillbuf] = rindex;
}

// P_NoiseAlert,
// If a monster yells at a player
// it will alert other monsters
void
P_NoiseAlert
( mob_t* target,
  mob_t* emitter )
{
    soundtarget = target;
    validcount++;
    P_RecursiveSound (emitter, sec);
}

// Range
boolean P_ChedMeleeRange
( mob_t* p;
  fixed_t dist;
```

Владимир Исмагилов  
**Месть программиста**

«ЛитРес: Самиздат»

2020

**Исмагилов В.**

Мечь программиста / В. Исмагилов — «ЛитРес: Самиздат»,  
2020

В отношениях между людьми, что важнее сила или умственные способности. Герой книги сталкивается с бывшим спутником теперь уже его жены. На что готов пойти "бывший" ради мести и "наказания" своего обидчика.

# Содержание

Глава 1. Происшествие	5
Конец ознакомительного фрагмента.	6

## Глава 1. Происшествие

Ночь. Телефонный звонок. Дежурный из числа технического персонала, сбивчиво пере-скакивая с мысли на мысль, попросил срочно прибыть на полигон. Я посмотрел на часы, стоящие на прикроватной тумбочке, 4.15 утра.

Вставать из кровати абсолютно не хотелось, но служба есть служба. Осторожно, чтобы не разбудить Екатерину, встал. В потемках вышел в коридор и только там включил свет. Из головы не уходила мысль, что там произошло такого, что потребовалось мое вмешательство. Начал вспоминать расписание занятий курсантов на сегодняшние сутки. Определенно занятий после 23.00 вечера не запланировано. Напротив, по плану ночью должны были начаться технические работы по обслуживанию комплекса. Я старший инструктор по подготовке пилотов, к техническим работам абсолютно никакого отношения не имею. Что же там начудили техники? Неопределенность ситуации нервировала.

Одевшись, спустился в гараж. Сел в автомобиль и задал конечную точку маршрута – комплекс подготовки пилотов. Машина плавно выехала из гаража и влилась в куцый в это время поток автомагистрали. Снизив прозрачность окон автомобиля, попытался уснуть.

Через тридцать минут я уже стоял перед дверями комплекса. Нажав на кнопку вызова, доложил дежурному, что прибыл по вызову. Использовать свой пропуск в это время суток я не мог. Прозвучал зуммер, дверь немного дрогнула, освобождаясь от удерживающей ее в закрытом состоянии силы электромагнита. Открыл дверь, вошел, кивнул в приветствии дежурному и направился по лестнице, ведущей в аппаратную комплекса.

– Ну, что у вас случилось, головастики?

Старший среди дежурной смены начал свой рассказ. Из рассказа выходило следующее.

По расписанию в десять вечера прибыла очередная группа курсантов первого курса на отработку маневрирования и стыковки со станцией в поясе астероидов. Длительность занятия один час. Через полчаса мониторы состояния здоровья у всех курсантов, подключенных к киберпространству, пришли в красную зону. Техники начали процедуру принудительного отключения от полигона. Но через несколько минут поняли, что отключения не происходит. Один из курсантов уже пребывал в коме, медики, прибывшие ранее, умчали пострадавшего в госпиталь.

– Сбой?

– Нет, проверку и обновление программа прошла сегодня. Протокол безопасности обновлен.

– Да вы и сами знаете, что без действующего протокола подключиться к программе не получится, – чуть с обидой протараторил техник.

Я задал уточняющий вопрос: – Где дежурный инструктор по подготовке пилотов?

## **Конец ознакомительного фрагмента.**

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.