将控制台任务拆分和修改

struct CONSOLE

{

    struct SHEET \*sht;

    int cur\_x, cur\_y, cur\_c;

};

void console\_task(struct SHEET \*sheet, unsigned int memtotal);                          // 控制台任务

void cons\_newline(struct CONSOLE \*cons); // 换行,当到达最后一行时还会自动滚动

void cons\_putchar(struct CONSOLE \*cons, int chr, char move);    // 显示

void cons\_runcmd(char \*cmdline, struct CONSOLE \*cons, int \*fat, unsigned int memtotal); // 命令行

void cmd\_mem(struct CONSOLE \*cons, unsigned int memtotal);                              // mem命令

void cmd\_cls(struct CONSOLE \*cons);   // cls命令

void cmd\_dir(struct CONSOLE \*cons);   // dir命令

void cmd\_type(struct CONSOLE \*cons, int \*fat, char \*cmdline);                           // type命令

在file中增加一个查找文件函数

struct FILEINFO \*file\_search(char \*name, struct FILEINFO \*finfo, int max); // 查找文件

将hlt.nas改成hello编写

加入

int cmd\_app(struct CONSOLE \*cons, int \*fat, char \*cmdline); // 文件执行命令

在汇编中增加一个远call来调用函数和显示字符API，以及C编写的API处理声明

GLOBAL \_farcall

GLOBAL \_asm\_hrb\_api

EXTERN \_hrb\_api

记得在.h中声明

编写三函数

void hrb\_api(int edi, int esi, int ebp, int esp, int ebx, int edx, int ecx, int eax); // 显示字符API处理

void cons\_putstr0(struct CONSOLE \*cons, char \*s);                                     // 没有长度根据'\0'结尾来输出字符

void cons\_putstr1(struct CONSOLE \*cons, char \*s, int l);                              // 有长度，输出字符

在

void init\_gdtidt(void)

中加入

 set\_gatedesc(idt + 0x40, (int)asm\_hrb\_api, 2 \* 8, AR\_INTGATE32);

修改Makefile

测试

