53struct.c

```
1  /* 53struct.c ファイルに保存されたデータを整形して構造体に読み込む
2
3      実行時引数にファイル「monter.dat」を指定して実行する
4  */
5  #include <stdio.h>
6  #include <stdlib.h>
7  #include <string.h>
8
9  #define MAX 21          /* モンスター名の最大長 */
10 #define OK          1    /* データチェックOK */
11 #define ERROR    0    /* データチェックNG */
12
13 /* 構造体_monsterを宣言 */
14 struct _monster{
15     char state;          /* データが正常か不正か（OK/ERROR） */
16     char name[MAX]; /* モンスター名 */
17     int attack;          /* 攻撃力 */
18     int defence;         /* 防御力 */
19     int hp;              /* HP */
20     int exp;             /* EXP */
21     int gold;            /* ゴールド */
22 };
23
24 /* 構造体_monster型を別名で宣言 */
25 typedef struct _monster MONSTER;
26
27 /* ----- モンスターデータを表示する関数 */
28 void showMonsterData(MONSTER monster){
29     if(monster.state == OK){
30         printf("%20s/攻撃力:%3d 守備力:%3d HP=%3d EXP=%3d G=%3d¥n", monster.name, mo
   nster.attack, monster.defence, monster.hp, monster.exp, monster.gold);     }
31 }
32
33 /* ----- メイン関数 */
34 int main(int argc, char *argv[])
35 {
36     int i;
37     FILE *fp;
38     MONSTER m[100];
39     int ret;
40     int monsters;       /* 読み込んだモンスターの数 */
41     int dataCount;          /* 読み込んだデータ数 */
42     int dataErrorCount;     /* エラーのあったデータ数 */
43     char buffer[256];
44
45     /* モンスターファイル名を指定して実行 */
46     if(argc != 2){
47         perror("読み込みファイル名を指定してください");
48         exit(1);
49     }
```

```c

	fp = fopen(argv[1], "r");
	if(fp == NULL){
		perror("ファイルオープンエラー");
		exit(1);
	}

	/* ファイル形式チェック */
	fgets(buffer, 256, fp);
	if(!strcmp(buffer, "MONSTER_FILE")){
		perror("モンスターファイルではありません");
		exit(1);
	}
	else{	/* 空行を読み込む */
		fgets(buffer, 256, fp);
	}

	/* モンスターデータを読み込む（3行目からがモンスターデータ） */
	i = 0;
	dataErrorCount = monsters = 0;
	while(fgets(buffer, 256, fp) != NULL){
		ret = sscanf(buffer, "%20s¥t%d¥t%d¥t%d¥t%d¥t%d¥n", m[i].name, &m[i].attack, &m[i].defence, &m[i].hp, &m[i].exp, &m[i].gold);
		if(ret == 6){ /* 6項目正常に読み込まれていればデータチェックOK */
			m[i].state = OK;
		}
		else{
			m[i].state = ERROR;
			printf("%d件目のデータにエラーがあります¥n", i+1);
			dataErrorCount ++;
		}
		i++;
	}
	dataCount = i;
	monsters = dataCount - dataErrorCount;
	printf("%dキャラクタ読み込みました¥n¥n", monsters);
	fclose(fp);

	/* 読み込んだキャラクタを全て表示 */
	for(i=0; i<dataCount; i++){
		showMonsterData(m[i]);
	}

	return 0;
}
```