概要

　アドバンストのリスト構造の参照をした。変数宣言を関数化した。

プログラム

|  |
| --- |
| #include<stdio.h>  #include<string.h>  //プロトタイプ宣言  int funcA(void);  int funcB(void);  int funcC(void);  void set(void);  struct TCB{//構造体宣言  struct TCB \*next;//構造変数宣言  int (\*task)();//関数ポインタ  }tcb[3];//構造体配列  int main(void){  struct TCB \*nowtcb;//構造体ポインタ  int count;//リスト数  set();//関数初期化  nowtcb=&tcb[0];//アドレスの指定  for(count=0;count<3;count++){//0~2でする  nowtcb->task();//関数の呼び出し  nowtcb=nowtcb->next;//次へ移行  }    }  void set(void){  tcb[0].task=funcA;//関数入力  tcb[0].next=&tcb[1];//ポインタ指定  tcb[1].task=funcB;  tcb[1].next=&tcb[2];  tcb[2].task=funcC;  }  int funcA(void){//関数  printf("funcA()\r\n"); //文字の出力  printf("7bjk1101\r\n");  }  int funcB(void){  printf("funcB()\r\n");  printf("inoue shunnsuke\r\n");  }  int funcC(void){  printf("funcC()\r\n");  printf("2019/10/17\r\n");  } |

