**Mesin Karakter**

(Model akses sekuensial)

Mesin karakter adalah mesin abstrak yang terdiri dari:

1. Pita yang berisi deret karakter yang di akhiri **‘.’** (titik) atau pita yang hanya berisi **‘.’** (titik) disebut pita kosong.
2. Tombol **START** dan **ADV**
3. Sebuah lampu **EOP** (End of Pita)
4. Jendela yang ukurannya sebesar 1 karakter untuk menunjukan posisi karakter yang sedang dibaca disebut dengan **CC** (Current Characther).

Mesin memiliki mekanisme untuk mengubah posisi pita dan menyalakan lampu EOP jika karakter yang berada pada jendela adalah karakter ‘.’ (titik). State mesin setiap saat ditentukan oleh CC dan lampu EOP. Tombol START dan ADV digunakan untuk mengubah state mesin. Mesin hanya dapat digunakan pada saat lampu EOP tidak menyala.







Primitif mesin karakter

|  |
| --- |
| **Procedure** START()  { I.S. sembarang }  { F.S. CC adalah karakter pertama pita }  { Jika CC==MARK, EOP menyala(true)}  { Jika CC!=MARK, EOP padam(false) } |

|  |
| --- |
| **Procedure** ADV()  { I.S. CC!=MARK }  { F.S. CC adalah karakter berikutnya CC pada I.S }  { Jika CC==MARK, EOP menyala(true) } |

|  |
| --- |
| **Function** EOP()🡪int  { mengirimkan true jika CC==MARK } |

Realisasi mesin karakter dalam bahasa C

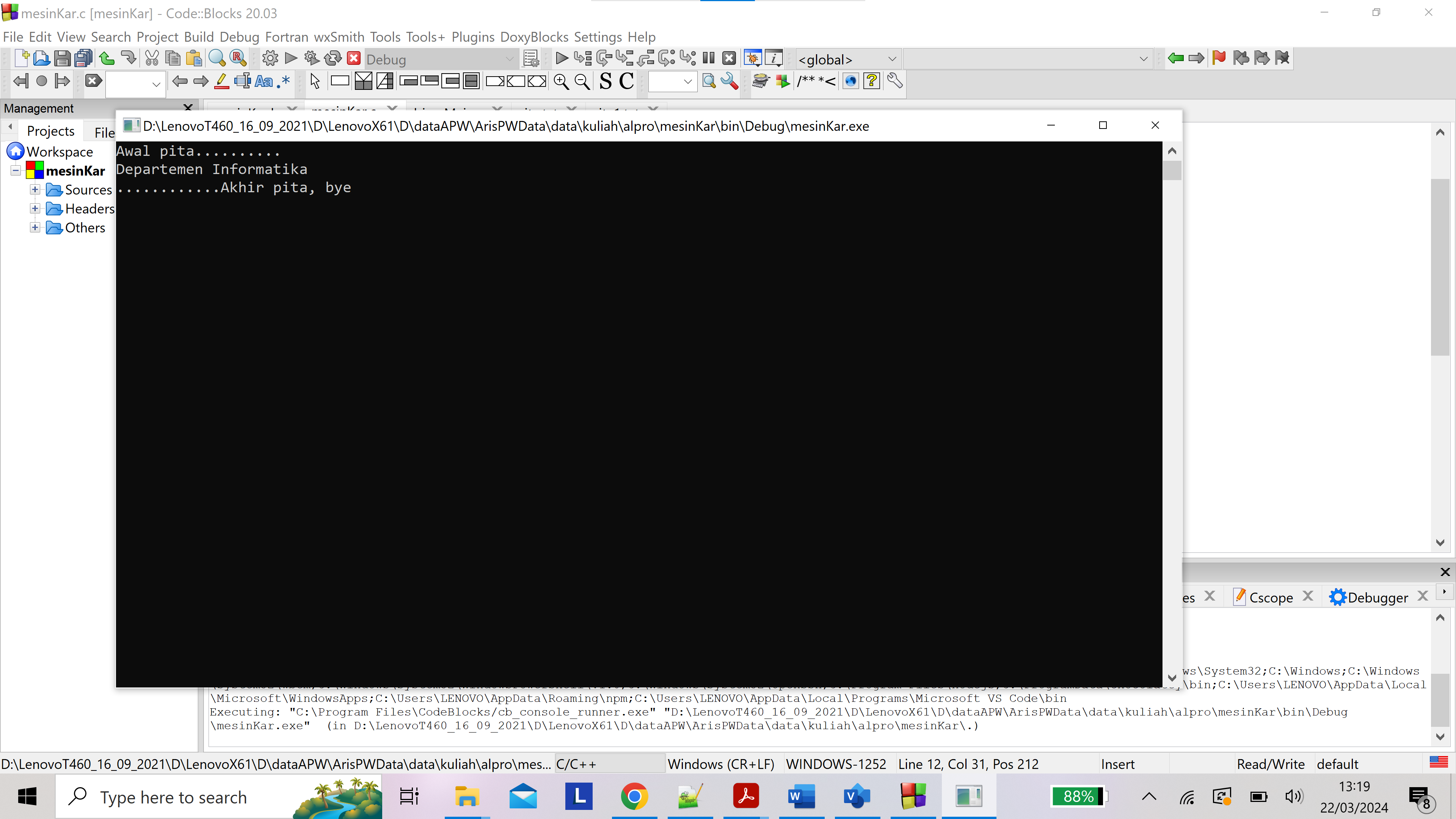
|  |
| --- |
| **/\*Nama file: mesinKar.h\*/**  **/\*Deskripsi: prototype mesin karakter\*/**  #ifndef MESINKAR\_H  #define MESINKAR\_H  #define False 0  #define True 1  #ifndef MARK  #define MARK '.'  #endif  #ifndef ENTER  #define ENTER 10  #endif  /\*Prototype\*/  void START();  /\* I.S. sembarang \*/  /\* F.S. CC adalah karakter pertama pita \*/  /\* Jika CC==MARK, EOP menyala(true) \*/  /\* Jika CC!=MARK, EOP padam(false) \*/  void ADV();  /\* I.S. CC!=MARK \*/  /\* F.S. CC adalah karakter berikutnya CC pada I.S \*/  /\* Jika CC==MARK, EOP menyala(true) \*/  int EOP();  /\* mengirimkan true jika CC==MARK \*/  #endif // MESINKAR\_H\_INCLUDED |
| **/\*Nama file: mesinKar.c\*/**  **/\*Deskripsi: body prototype mesin karakter\*/**  #ifndef MESINKAR\_C  #define MESINKAR\_C  #include <assert.h>  #include <fcntl.h>  #include <stdio.h>  #include "mesinKar.h"  /\* definisi states \*/  char CC;  /\* definisi Pita \*/  char Pita\_karakter[100]="pita.txt";  static FILE \*FILEKU;  static int retval;  /\*Body Prototype\*/  void START()  /\* I.S. sembarang \*/  /\* F.S. CC adalah karakter pertama pita \*/  /\* Jika CC==MARK, EOP menyala(true) \*/  /\* Jika CC!=MARK, EOP padam(false) \*/  {  FILEKU=fopen(Pita\_karakter,"r");  retval=fscanf(FILEKU,"%c",&CC);  }  void ADV()  /\* I.S. CC!=MARK \*/  /\* F.S. CC adalah karakter berikutnya CC pada I.S \*/  /\* Jika CC==MARK, EOP menyala(true) \*/  {  retval=fscanf(FILEKU,"%c",&CC);  if(CC==MARK)  {  fclose(FILEKU);  }  }  int EOP()  /\* mengirimkan true jika CC==MARK \*/  {  return(CC==MARK);  }  #endif |

|  |
| --- |
| **/\*Nama file: driverMesinKar.c\*/**  **/\*Deskripsi: driver mesin karakter\*/**  #include <assert.h>  #include <fcntl.h>  #include <stdio.h>  #include "mesinKar.h"  /\* definisi states secara global\*/  char CC;  int main(){  /\*kamus\*/  /\*algoritma\*/  printf("Awal pita..........\n");  START();  while (!EOP())  {  printf("%c",CC);  ADV();  }  printf("\n............Akhir pita, bye\n");  return 0;  } |

Pita.txt

|  |
| --- |
| Departemen Informatika. |

Hasil eksekusi



**Soal latihan dan Penugasan**

1. Diberikan sebuah file pitaKarakter.txt sembarang yang dibaca sebagai sebuah file, maka buatlah teks algoritma sebagai berikut:
2. Menentukan banyaknya jumlah semua karakter.
3. Menentukan banyaknya vokal, dan (konsonan dan karakter lainnya).
4. Menentukan banyaknya kata, untuk definisi kata silahkan anda gunakan asumsi masing-masing (misal asumsi kata adalah setiap ketemu spasi, koma, … adalah dihitung sebagai kata).
5. Menentukan banyaknya pasangan karakter “LE”, misal pitaKarakter berisikan SAYA **LE**BIH SUKA MAKAN **LELE** **LE**ZAT. Pasangan “LE” ada sebanyak 4.
6. Diberikan sebuah file text HTML sebarang yang dibaca dari sebuah file (sebagai contoh pada file source.html) dibawah ini:

|  |
| --- |
| <html>  <head>  <title>Page Title</title>  </head>  <body>  <h1>My First Heading</h1>  <p>My first paragraph.</p>  </body>  </html> |

1. Buatlah teks algoritma untuk menentukan (menghitung) banyaknya tag-tag HTML yang terdapat pada file text HTML. Sebagai contoh untuk pitakarakter.txt seperti pada contoh bagian di atas, maka akan menghasilkan sebanyak 6 tag HTML.
2. Buatlah teks algoritma untuk menampilkan tag-tag HTML yang terdapat pada file text HTML. Sebagai contoh untuk pitakarakter.txt seperti pada contoh bagian di atas, maka akan menghasilkan tag HTML yang terdiri dari: html, head, title, body, h1, dan p.