

11

Датотеки

1. Да се напише програма која за дадена текстуална датотека “text.txt” ќе ги изброи зборовите. Зборовите се состојат од алфанумерички знаци и се одделени со најмалку едно празно место или знаци за interpunkcija.

```
#include <stdio.h>
int AlphaNum(char k)
{
    int mala,golema,cifra;

    mala  =(k>='a') && (k<='z');
    golema=(k>='A') && (k<='Z');
    cifra =(k>='0') && (k<='9');
    return(mala || golema || cifra);
}
main()
{
    char  c;
    int   brzborovi, vozbor;
    FILE  *dat;

    brzborovi=0; vozbor=0;
    if((dat=fopen("text.txt","r"))==NULL)
    {
        fprintf(stderr,"Ne mozam da ja otvoram datotekata text.txt");
        return(-1);
    }
    while((c=fgetc(dat))!=EOF) /* citaj znak po znak se do krajot */
    {
        if (AlphaNum(c))
        {
            if (!vozbor) vozbor=1;
        }
        else
            if (vozbor) /* ako do sega sme bile vo zbor - izbroj go */
            {
                vozbor=0;
                brzborovi++;
            }
    }
    if (vozbor) brzborovi++; /* proveri da ne zavrсуva na bukva */
    printf("Vкупно %d zborovi\n",brzborovi);
    return(0);
}
```

2. Да се напише програма која за дадена текстуална датотека, чие што име се внесува од командната линија, ќе го одреди односот на самогласките и согласките.

```
#include <stdio.h>
int Bukva(char k)
{
    int mala,golema;

    mala  =(k>='a') && (k<='z');
    golema=(k>='A') && (k<='Z');
    return(mala || golema);
}
int Samoglaska(char k)
{
    switch (k){
        case 'a':
        case 'e':
        case 'i':
        case 'o':
        case 'u':
        case 'A':
        case 'E':
        case 'I':
        case 'O':
        case 'U': return(1);
        default:  return(0);
    }
}
main(int argc, char *argv[])
{
    char  c;
    int   brsogl=0, brsamogl=0;
    FILE  *dat;
    if((dat=fopen(argv[1],"r"))==NULL)
    {
        fprintf(stderr,"Ne mozam da ja otvoram datotekata %s",argv[1]);
        return(-1);
    }
    while((c=fgetc(dat))!=EOF) /* citaj znak po znak se do krajot */
    {
        if (Bukva(c))
        {
            if (Samoglaska(c)) brsamogl++;
            else brsogl++;
        }
    }
    printf("Odnosot samoglaski/soglaski = %d/%d = %5.2f\n",
           brsamogl, brsogl, (float) brsamogl/brsogl);
    return(0);
}
```

3. За дадена датотека, чие што име се задава од командна линија, да се испечати вкупниот број на зборови кои почнуваат и завршуваат со иста буква. Да се провери дали името на датотеката е зададено преку командната линија.

```
#include <stdio.h>
int Bukva(char k)
{ int mala,golema;
  mala  =(k>='a') && (k<='z');
  golema=(k>='A') && (k<='Z');
  return(mala || golema);
}

main(int argc, char *argv[])
{ char c, prva, posledna;
  int  brzborovi=0, vozbor=0;
  FILE *dat;
  if (argc==1) {
    printf("Ne e zadadeno imeto na datotekata\n");
    return(-1);
  }
  else
  {
    if((dat=fopen(argv[1],"r"))==NULL)
    { printf("Ne mozam da ja otvoram datotekata %s\n",argv[1]);
      return(-1);
    }
    while((c=fgetc(dat))!=EOF) /* citaj znak po znak se do krajot */
    {
      if (Bukva(c))
      {
        if (!vozbor)
        {
          vozbor=1;
          prva=c;
        }
        posledna=c; /* posledna kje bide sekoja буква od zborot */
      }
      else
      {
        if (vozbor) /* ako do sega sme bile vo zbor */
        {
          vozbor=0;
          if (prva==posledna) brzborovi++; /* togas proveruvame */
        }
      }
      if (vozbor && (prva==posledna)) brzborovi++; /* za na kraj */
      printf("%d zborovi pocnuvaat i zavrshuvaat na ista буква\n",
        brzborovi);
    }
    return(0);
  }
}
```

4. Da se napiše programa koja za dadena tekst datoteka }e izbroi kolku zborovi podolgi od 3 bukvi po~nuvaat i završevaat na ista bukva. Da ne se pravi razlika meju golema i mala bukva. Zborovite se sostaveni od proizvolen broj na bukvi, a meju sebnost se oddeleni so najmalku eden specijalen znak, cifra ili belina. Imeto na vlezna datoteka se zadava od komandna linija, a dokolku ne e zadadeno se ~ita od standardniot vlez.

```
#include <stdio.h>
#include <ctype.h> /* zaradi funkcii isalpha i toupper */
main(int argc, char *argv[])
{
    char prva=0, posledna=0, c;
    int b=0; /* brojac na zborovi */
    int vo_zbor=0; /* oznaka deka se naogame vo zbor */
    int len=0; /* dolzina na zborot */
    FILE *f_vlezna;
    if(argc>2) /* ako ime poveke od eden parametar vo komandnata linija */
    {
        /* ispecati kako se upotrebuva programata */
        printf("Upotreba: %s ime_na_datoteka\n", argv[0]);
        return(-1);
    }
    if(argc==1) /* ako ne e zadaden parametar vo komandnata linija */
        f_vlezna=stdin; /* citaj od standardniot vlez završi so Ctrl+Z */
    else if((f_vlezna=fopen(argv[1], "r"))==NULL)
    {
        /* inaku od datotekata zadadena vo komandnata linija */
        fprintf(stderr, "Ne mozam da ja otvoram datotekata %s\n", argv[1]);
        return(0);
    }
    while((c=fgetc(f_vlezna))!=EOF) /* citaj znak po znak se do krajot */
    {
        if(isalpha(c)) /* ako e procitana bukva */
        {
            if(!vo_zbor) /* ako ne se naogas vo zbor */
            {
                vo_zbor=1; /* odbelezi deka si vlegol vo zbor */
                prva=c; /* ova e voedno i prvata bukva od zborot */
            }
            len++; /* izbroj ja stotuku procitanata bukva */
            posledna=c; /* zapomni ja poslednata procitana bukva */
        }
        else /* ako procitanoto ne e bukva */
        {
            if(vo_zbor) /* ako si se naogal vo zbor */
            {
                vo_zbor=0; /* odbelezi deka si izlegol od zbor */
                if(len>3 && toupper(prva)==toupper(posledna))
                {
                    b++; /* dokolku zborot gi zadovoluva uslovite izbroj go */
                    len=0; /* resetiraj ja dolzinata za merenje na sledniot zbor */
                }
            }
        }
    }
    if(vo_zbor) /* ako si se naogal vo zbor na krajot od datotekata */
    {
        if(len>3 && toupper(prva)==toupper(posledna))
        {
            b++; /* dokolku zborot gi zadovoluva uslovite izbroj go */
            len=0; /* resetiraj ja dolzinata za merenje na sledniot zbor */
        }
    }
    printf("\nIma %d takvi zborovi\n", b);
}
```

5. Да се напише програма која ќе ги прочита елементите од една матрица сместени во текстуалната датотека "matrical.txt". На почетокот на датотеката се запишани бројот на редици и колони на матрицата. Секој елемент е реален број запишан во посебен ред во датотеката. Потоа матрицата треба да се транспонира и да се запише во нова датотека "matrica1.txt" на истиот начин.

```
#include <stdio.h>
#define MAX 10
main()
{ int    i,j,m,n;
  float a[MAX][MAX], b[MAX][MAX];
  FILE   *vleзна, *izleзна;

  if((vleзна=fopen("matrical.txt","r"))==NULL)
  { printf("Ne mozam da ja otvoram datotekata matrical.txt");
    return(-1);
  }
  if (!feof(vleзна)) fscanf(vleзна, "%d", &m);
  if (!feof(vleзна)) fscanf(vleзна, "%d", &n);
  if ((m>MAX) || (n> MAX))
  {
    printf("Mnogu golema matrica!");
    return(-1);
  }
  for (i=0; (i<m) && !feof(vleзна) ; i++)
    for (j=0; (j<n) && !feof(vleзна) ; j++)
      fscanf(vleзна, "%f", &a[i][j]);

  fclose(vleзна);
  if ((i!=m) || (j!=n))
  {
    printf("Nema dovolno podatoci vo datotekata!");
    return(-1);
  }
  for (i=0;i<m;i++)
    for (j=0;j<n;j++)
      b[j][i]=a[i][j];
  if((izleзна=fopen("matrica2.txt","w"))==NULL)
  {
    printf("Ne mozam da ja otvoram datotekata matrica2.txt");
    return(-1);
  }
  fprintf(izleзна, "%d\\n", n); /* obratno */
  fprintf(izleзна, "%d\\n", m);
  for (i=0;i<n;i++)
    for (j=0;j<m;j++)
      fprintf(izleзна, "%7.2f\\n", b[i][j]);
  fclose(izleзна);
  return(0);
}
```

6. Дадена е текстуална датотека 'text.txt'. Да се напише програма која ќе го отпечати бројот на редови во кои има по најмногу десет зборови. Зборовите се составени само од букви. Во секој ред има најмногу 80 знака.

```
#include <stdio.h>
#define MAX 80
int AlphaNum(char k)
{   int mala,golema,cifra;

    mala  =(k>='a') && (k<='z');
    golema=(k>='A') && (k<='Z');
    cifra =(k>='0') && (k<='9');
    return(mala || golema || cifra);
}
main()
{   char  s[MAX],c;
    int i,  zbor, red=0, vozbor;
    FILE  *dat;

    if((dat=fopen("text.txt","r"))==NULL)
    {
        printf("Ne mozam da ja otvoram datotekata text.txt");
        return(-1);
    }

    while((fgets(s, MAX, dat))!=NULL) /* red po red, NULL za kraj */
    {
        i=0; c=s[0]; zbor=0; vozbor=0;
        while (c!='\n')
        {
            if (AlphaNum(c))
            {
                if (!vozbor) vozbor=1;
            }
            else
                if (vozbor) /* ako do sega sme bile vo zbor - izbroj go */
                {
                    vozbor=0;
                    zbor++;
                }
            i++; c=s[i];
        }
        if (vozbor) zbor++; /* proveri da ne zavrсуva na bukva */
        if (zbor > 10) red++;
    }
    printf("Vkupno %d redovi imaat povekje od 10 zborovi\n",red);
    return(0);
}
```