Rešitve pisnega izpita z dne 7. septembra 2001

1. naloga (25%)

```
#include <stdio.h>
#define MAX 26

main() {
  int c, i;
  int crke[MAX];

  for( i=0; i<MAX; i++ )
    crke[i] = 0;
  while( (c=getchar()) != EOF ) {
    if( c != '\n' ) {
      if( (c >= 'a') && (c <= 'z') )
        ++crke[c-'a'];
      if( (c >= 'A') && (c <= 'Z') )
        ++crke[c-'A'];
    }
    else {
      for( i=0; i<MAX; i++ ) {
        if( crke[i] > 0 )
          printf("%c ",'a'+i);
        crke[i] = 0;
      }
      printf("\n");
    }
  }
}
```

2. naloga (25%)

```
#include <stdio.h>
#include <ctype.h>

main(int argc, char *argv[]) {
  FILE *fp1, *fp2;
  int ch, prev;

  if( (fp1 = fopen(argv[1], "r")) == NULL)
    exit(0);
  if( (fp2 = fopen(argv[2], "w")) == NULL) {
    fclose(fp1);
    exit(0);
  }
  prev = 0;
  while( (ch=getc(fp1)) != EOF ) {
    if( isalpha(ch) && ((prev == ' ') || (prev == '\n') || (prev == '\t')) )
      putc(toupper(ch), fp2);
    else
      putc(tolower(ch), fp2);
    prev = ch;
  }
  fclose(fp1);
  fclose(fp2);
}
```

3. naloga (25%)

```
#include <stdio.h>
#include <math.h>
#define MAX_TOCK 10
```

```
#define MAX_CRT 10

int tocke[MAX_TOCK][2];
int crte[MAX_CRT][2];

main() {
  int i;
  int x1, y1, x2, y2;
  double dolzina;

  dolzina = 0;
  for( i=0; i<MAX_CRT; i++ ) {
    x1 = tocke[crte[i][0]][0];
    y1 = tocke[crte[i][0]][1];
    x2 = tocke[crte[i][1]][0];
    y2 = tocke[crte[i][1]][1];
    dolzina += sqrt((y2-y1)*(y2-y1) + (x2-x1)*(x2-x1));
  }
}
```

4. naloga (25%)

```
void dodaj(struct elem **p, char niz[]) {
  struct elem *q, *r;

  q = (struct elem*) malloc(sizeof(struct elem));
  strcpy(q->niz, niz);
  if ( (*p == NULL) || (strcmp(niz, (*p)->niz) < 0) ) {  // dodaj na zacetek seznama
    q->naprej = *p;
    q->nazaj = NULL;
    (*p)->nazaj = q;
    *p = q;
  } else {
      // poisci ustrezno mesto v seznamu za nov element
      r = *p;
      while ( (r->naprej != NULL) && (strcmp(niz, r->naprej->niz) > 0 ) )
        r = r->naprej;
      // dodaj za element, na katerega kaze r
      q->naprej = r->naprej;
      r->naprej = q;
      q->nazaj = r;
      if( q->naprej != NULL)
        q->naprej->nazaj = q;
  }
}
```