

Rešitve pisnega izpita (programski jeziki Java in C) z dne 9. junija 2004

1. naloga (25%)

```
public int pojavitve(String niz, char znak) {
    int stevec = 0;
    for(int i=0; i<niz.length(); i++) {
        if(niz.charAt(i) == znak)
            stevec ++;
    }
    return(stevec);
}
```

2. naloga (25%)

```
class Pisalec {
    private String imeDat;

    Pisalec(String ime) {
        imeDat = ime;
    }

    public void pisiInt(int stevilo) throws IOException {
        BufferedWriter izhod = new BufferedWriter(new FileWriter(imeDat, true));
        izhod.write(Integer.toString(stevilo));
        izhod.close();
    }

    public void pisiString(String niz) throws IOException {
        BufferedWriter izhod = new BufferedWriter(new FileWriter(imeDat, true));
        izhod.write(niz);
        izhod.close();
    }
}
```

3. naloga (25%)

```
#include <stdio.h>

main(int argc, char *argv[]) {
    FILE *fp;
    int znak, pred;

    fp = fopen(argv[1], "r");
    pred = '\n';
    while( (znak=getc(fp)) != EOF ) {
        if( (pred == '\n') || (pred == '\t') || (pred == ' ') ) {
            if( (znak >= 'a') && (znak <= 'z') )
                znak = znak - ('a' - 'A');
        }
        putchar(znak);
        pred = znak;
    }
    fclose(fp);
}
```

4. naloga (25%)

```
struct elem {
    int stevilo;
```

```
    struct elem *naprej;
};

struct elem *presek(struct elem *p, struct elem *q) {
    struct elem *r, *t;

    r = NULL;
    if( (p == NULL) || q == NULL)
        return(r);
    while(p != NULL) {
        t = q;
        while(t != NULL) {
            if(p->stevilo == t->stevilo) {
                dodaj(&r, p->stevilo);
                break;
            }
            t = t->naprej;
        }
        p = p->naprej;
    }
    return(r);
}

void dodaj(struct elem **zacetek, int st) {
    struct elem *s, *t;
    if(*zacetek == NULL) {
        s = (struct elem*) malloc(sizeof(struct elem));
        s->stevilo = st;
        s->naprej = NULL;
        *zacetek = s;
    }
    else {
        t = *zacetek;
        while(t != NULL) {
            if(t->stevilo == st) // stevilo ze obstaja v seznamu
                return;
            t = t->naprej;
        }
        s = (struct elem*) malloc(sizeof(struct elem));
        s->stevilo = st;
        s->naprej = *zacetek; // nov element dodamo na zacetek seznama
        *zacetek = s;
    }
}
```