

Ohjeita

Kun tentti järjestetään kahdessa paikassa samanaikaisesti, ei tentin valvoja voi vastata tenttiä koskeviin kysymyksiin vaan tenttiin on vastattava sellaisenaan, mahdollisine virheineenkin.

Tavoite on, että kirjoittamasi ohjelmat/funktiot/jne olisivat kääntäjälle kelpavaa C-kieltä. Eli

- C-kielessä ei ole "jne", "...", "--||--", jne merkintöjä!
- Kommentit kommentteihin /* selitys */.
- C-kielen avainsanat ja varusfunktiot kirjoitetaan pienaakkosilla (ei siis INT, vaan int).
- Välilyönti on selkeä tyhjä tila kahden merkin välissä (eikä "␣" tms).
- Kirjoita selvästi, ettei tarkastaja tulkitse väärin.
- Pienet syntaksivirheet eivät kuitenkaan juuri pudota arvostelua.

Työskentelystä

- Kuvan piirtäminen helpottaa ratkaisun kirjoittamista.
- Suositus: hahmottele ratkaisu ensin suttupaperille, aloita varsinaisen vastauksen kirjoitus vasta sitten.
- Yksinkertaisessa ratkaisussa on yleensä vähemmän virheitä kuin monimutkaisessa.
- Selkeässä ratkaisussa rajatapaukset toimivat yleensä paremmin kuin kikkailuratkaisussa.
- Lue annettu tehtävä kunnolla, väärän asian toteuttaminen ei kelpaa, vaikka selitys olisi väärinymmärrys.

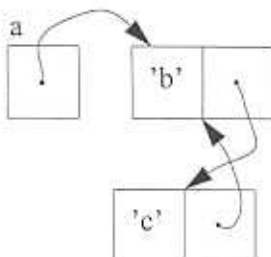
1) Anna seuraavien C-kokonaisuuksien esittely (funktioistakin vain esittely, ei toteutusta). (10 p)

- Merkkijonotaulukko (10 merkkijonoa).
- Liukulukutaulukon keskiarvon palauttava funktio.
- Rakenne päiväys (päivä, kuukausi, vuosi, viikonpäivä).
- Päiväystaulukko (10 hloä, edellisen tehtävän päiväykset).
- Edellisen tehtävän mukaisen päiväystaulukon luova ja palauttava funktio.

2) a) *Piirrä kuva* seuraavien C-kielisten esittelyiden ja lauseiden mukaisesta rakenteesta muistissa. (6p)

```
typedef struct a {
    int b;
    int *c;
} d;
int *e;
d *f = (d*)malloc(3*sizeof(struct a));
f[0].b = 1;
f[2].c = (int*)malloc(3*sizeof(int));
e = f[2].c;
*(e+1) = 2;
```

b) Anna C-kieliset *esittelyt ja lauseet* seuraavan kuvan mukaisen rakenteen muodostamiseksi. (6p)



3) Kirjoita C-kielinen funktio

```
int koestrlcat(char *kohde, char *lahde, int n);
```

joka lisää merkkijonon *lahde* sisällön merkkijonon *kohde* loppuun, kuitenkin niin, että tuloksena syntyvä yhdiste merkkijonossa *kohde* vie tilaa enintään *n* tavua (terminointi mukaanlukien). Jollei koko merkkijonoa *lahde* voitu lisätä merkkijonon *kohde* perään, merkkijonosta *lahde* lisätään vain ne merkit alusta jotka mahtuvat.

Funktio palauttaa tuloksenaan alkuperäisten merkkijonojen *kohde* ja *lahde* yhteispituuden. Jos syötteet *kohde* ja *lahde* mahtuivat tulokseen *kohde*, palautusarvo on sama kuin uuden merkkijonon *kohde* pituus. Jos merkkijonon *lahde* kaikki merkit eivät mahtuneet merkkijonon *kohde*, palautusarvo on edelleen alkuperäisten merkkijonojen *kohde* ja *lahde* yhteispituus ja siten suurempi tai yhtäsuuri kuin *n*. Muista terminoinnit sekä syötteissä *kohde* ja *lahde*, että valmiissa tuloksessa *kohde*.

Ratkaisussa *ei saa* käyttää valmiita *str..()* funktioita. Piirrä kuva funktion toiminnasta. (13p)