

TÖL203G Tölvunarfræði 2

Miðmísserispróf 2012

Öll dæmin hafa sama vægi. **Aðeins þarf að leysa 7 dæmi af 10. Sjö bestu dæmin gilda.** Engin skrifleg hjálpargögn leyfileg.

1. Helmingunarleit. Fyllið inn þar sem spurningamerkin eru, þ.e. skrifið hvað á að koma í stað ?1?, o.s.frv. Þetta eru samtals átta svör.

```
// Notkun: k = leita(f,i,j,x);
// Fyrir: f[i..j-1] er í vaxandi röð
// Eftir: f[i..j-1] er óbreytt, i <= k <= j, og
//      f[i..k-1] < x <= f[k..j-1]
int leita(double[] f, int i, int j, double x) {
    if (i == j) return i;
    int m = (i+j)/2;
    if (f[m] ?1? x)
        return leita(f,i,?2?,x);
    else
        return leita(f,?3?,j,x);
}
```

```
int leita(double[] f, int i, int j, double x) {
    int p = i, q = j;
    while (?4?) {
        // | <x | óþekkt | >=x |
        // i      p          q      j
        int m = (?5?)/2;

        if (f[m] ?6? x)
            p = ?7?;
        else
            q = ?8?;
    }
    return p;
}
```

2. Er eftirfarandi útfærsla á valröðun (selection sort) stöðug? Rökstyðjið eða sýnið mótdæmi.

```
// Notkun:  ssort(f,a,b);
// Fyrir:   f[i..j-1] er svæði í f
// Eftir:   Búið er að raða svæðinu f[i..j-1] í vaxandi röð.
//          Röðunaraðferðin er selection sort.
static void ssort(Comparable[] f, int i, int j)
{
    for (int p = i; p < j; p++)
    {
        // f[i..p-1] inniheldur minnstu gildi úr f[i..j-1]
        // í vaxandi röð
        int min = p;
        for (int k = p+1; k < j; k++)
        {
            // f[i..p-1] inniheldur minnstu gildi úr f[i..j-1]
            // í vaxandi röð
            // f[min] <= f[p..k-1]
            if (less(f[k], f[min]))
                min = k;
        }

        swap(f, p, min);
    }
}
```

3. Hver er fastayrðing ytri lykkju í insertion sort? Hver er tímaflækja insertion sort í versta falli? En í besta falli?
4. Gerið grein fyrir meginreglunni um upplýsingahuld: Hver er reglan og hver er tilgangur hennar?
5. Eftirfarandi runu af gildum er bætt í tóma hrúgu með stærsta gildi efst: 40 10 30 20 50. Teiknið mynd af hrúgunni eftir að öll gildin hafa verið sett í hana. Þið þurfið ekki að sýna stöðuna í hverju skrefi heldur bara lokastöðuna eftir að öll gildin hafa verið sett í hrúguna.

6. Gefið er eftirfarandi fylki sem inniheldur hrúgu með stærsta gildi efst:

```
int[] f = {100,50,80,30,10,50,20};
```

Teiknið mynd af hrúgunni eins og hún er í upphafi. Fjarlægið síðan stærsta stakið úr henni og teiknið mynd af hrúgunni eftir það.

7. Lýsið hugmyndinni í mergesort. Hver er tímaflækja mergesort?
8. Skrifðu fastayrðingu gagna fyrir útfærslu á hlaða og skrifðu notkunarlýsingu ásamt útfærslu fyrir Push aðferðina.
9. Skrifðu fall í Java sem fær sem inntak bendi á haus í eintengdum lista heiltalna og skrifar út allar tölurnar.

Skilgreining á hlekk:

```
static class Node
{
    int value;
    Node next;
};
```

Haus fyrir fallið:

```
static void print(Node head)
{
    // ..
}
```

10. Gerum ráð fyrir eftirfarandi fylki af heiltölum:

```
int[] f = {5, 3, 8, 1, 2, 6, 4, 9, 7}
```

Gerum síðan ráð fyrir að fylkinu sé skipt með quicksort partition falli þar sem $f[0] == 5$ er notað sem vendistak (e. pivot).

Hvað gildir um fylkið eftir skiptinguna?