Ασκήσεις (1)

1) Προσδιορίστε τον χρόνο εκτέλεσης με βάση το λεπτομερές μοντέλο υπολογιστή που περιγράψαμε στη θεωρία (βλ. διαφάνειες «Εισαγωγή στην Ανάλυση Αλγορίθμων (1)) για κάθε ένα από τα παρακάτω αποσπάσματα προγράμματος.

(i)
```
for (unsigned int i = 0; i < n; ++i)
    ++k;
```

(ii)
```
for (unsigned int i = 1; i < n; i *= 2)
    ++k;
```

(iii)
```
for (unsigned int i = n - 1; i != 0; i /= 2)
    ++k;
```

(iv)
```
for (unsigned int i = 0; i < n; ++i)
    if (i % 2; == 0)
        ++k
```

(v)
```
for (unsigned int i = 0; i < n; ++i)
    for (unsigned int j = 0; j < n; ++j)
        ++k;
```

(vi)
```
for (unsigned int i = 0; i < n; ++i)
    for (unsigned int j = i; j < n; ++j)
        ++k;
```

(vii)
```
for (unsigned int i = 0; i < n; ++i)
    for (unsigned int j = 0; j < i * i; ++j)
        ++k;
```

2) Να επαναλάβετε την Άσκηση 1 με βάση αυτή τη φορά το απλοποιημένο μοντέλο υπολογιστή που περιγράψαμε στη θεωρία (βλ. διαφάνειες «Εισαγωγή στην Ανάλυση Αλγορίθμων (1)).

3) Γράψτε ένα μη αναδρομικό πρόγραμμα για τον υπολογισμό του παραγοντικού ενός φυσικού αριθμού n με βάση τον τύπο

$$n! = \begin{cases} 1, & n = 0 \\ \prod_{i=1}^{n} i, & n > 0 \end{cases}$$

Υπολογίστε στη συνέχεια τον χρόνο εκτέλεσης που προβλέπεται από το λεπτομερές και από το απλοποιημένο μοντέλο υπολογιστή.

4) Θεωρούμε το *πρόβλημα μοναδικότητας στοιχείου* σύμφωνα με το οποίο ελέγχουμε αν τα στοιχεία μιας συστοιχίας είναι διακεκριμένα (όλα διαφορετικά μεταξύ τους). Ένας τρόπος επίλυσης του προβλήματος αυτού είναι ο ακόλουθος αλγόριθμος (σε μορφή ψευδοκώδικα):

Αλγόριθμος ΔιακεκριμέναΣτοιχεία(A[0..n – 1]
//Είσοδος: Συστοιχία A[0..n – 1]
//Εξοδος: “True” αν όλα τα στοιχεία είναι διακεκριμένα και “False” διαφορετικά.

```
for i ← 0 to n – 2
    for j ← i + 1 to n – 1
        if A[i] = A[j] return False
return True
```

Θεωρώντας ως βασική πράξη (βαρόμετρο) την πράξη της σύγκρισης που κάνει ο αλγόριθμος στον εσωτερικό βρόχο, προσδιορίστε σύμφωνα με το απλοποιημένο μοντέλο τον χρόνο εκτέλεσης χειρότερης περίπτωσης του αλγορίθμου.

5) Για κάθε ένα από τα τμήματα αλγορίθμων των περιπτώσεων (1) – (11), υποθέστε ότι το n είναι ένας θετικός ακέραιος.

i) Υπολογίστε το πλήθος $S(n)$ των αριθμητικών πράξεων (προσθέσεων, αφαιρέσεων και πολλαπλασιασμών) που πρέπει να γίνουν κατά την εκτέλεση του τμήματος του αλγορίθμου.

ii) Εκφράστε την πολυπλοκότητα $S(n)$ του τμήματος του αλγορίθμου στον συμβολισμό O.

(1)

for $i := 3$ **to** $n - 1$
 $a := 3 \cdot n + 2 \cdot i - 1$

(2)

for $i := 1$ **to** $\lfloor n/2 \rfloor$
 $a := n - i$

(3)

for $i := 1$ **to** n
 for $j := 1$ **to** $2n$
  $a := 2 \cdot n + i \cdot j$

(4)

for $k := 2$ **to** n
 for $j := 1$ **to** $3n$
  $x := a[k] - b[j]$

(5)

for $k := 1$ **to** $n - 1$
 for $j := 1$ **to** $k + 1$
  $x := a[k] + b[j]$

(6)

$t := 0$
for $i := 1$ **to** n
 $s := 0$
 for $j := 1$ **to** i
  $s := s + a[j]$
 $t := t + s^2$

(7)

$r := 0$
for $i := 1$ **to** n
 $p := 1$
 $q := 1$
 for $j := i + 1$ **to** n
  $p := p \cdot c[j]$
  $q := q \cdot c[j]^2$
 $r := p + q$

(8)

$t := 0$
for $k := 1$ **to** n
    $s := 0$
    for $j := 1$ **to** $i - 1$
        $s := s + j \cdot (i - j + 1)$
    $r := s^2$

(9)

for $i := 1$ **to** n
    for $j := 1$ **to** $\lfloor (i + 1)/2 \rfloor$
        $a := (n - i) \cdot (n - j)$

(10)

for $i := 1$ **to** n
    for $j := 1$ **to** $2n$
        for $k := 1$ **to** n
            $x := i \cdot j \cdot k$

(11)

for $i := 1$ **to** n
    for $j := 1$ **to** i
        for $k := 1$ **to** j
            $x := i \cdot j \cdot k$