# Složenost

1. Odrediti red slozenosti i obrazloziti odgovor.

s=0;

for(i=1;i<=n;i++)

if(i% (int) sqrt(n)==0)

for(j=0;j<=i\*i;j++)

s+=j\*j;

2. Red slozenosti sledeceg koda je n\*2n,2n ili n2 , obrazloziti odgovor.

i=1;s=0;

while(i<=n)

{

k=i; i++; j=n;

while(j>k)

{

j--; i\*=2;

}

for(k=1;k<i;k++)

s+=i;

}

3. Odredi red slozenosti i obrazlozi odgovor.

i=j=0;

while(i<n && j<n)

{

if(a[i]<=b[j])

{

c[k++]=a[i++];

while(i<n && a[i-1]==a[i]) i++;

}

else

{

c[k++]=b[j++];

while(j<n && b[j-1]==b[j]) j++;

}

}

4. Odredi u funkciji od n red slozenosti.

i=n/8; s=0;

while(i<n)

{ for(k=1;k<=n;k++)

{ j=k;

while(j<n)

{ j++; s+=i; }

}

i=2\*i;

}

5. Odredi red slozenosti u funkciji od n. (n\*logn, n log n ili (logn) n)

i=k=1;

while(i<=n)

{ i\*=2; k++;}

r=1;s=0;

for(i=0;i<=n;i++)

{

for(j=1;j<=r;j++)

s+=j\*i;

r\*=k;

}

6. Odredi red slozenosti u funkciji od n. ( nlogn, 2n ili n\*2n)

i=0; s=1;

while(i<n)

{

i++; k=s;

for(j=1;j<=s;j++)

k++;

s=k;

}

7. Odredi red slozenosti u funkciji od n.

for (i=1;i<=n;i++)

for (j=1;j<=i\*i;j++)

{ k=1; m=n;

while (m>=k)

{

k=k\*3;

m=m/2;

}

}

8. Odredi red slozenosti u funkciji od n.

s =k= 0;

for (i= 1;i<=n;i++)

{

k= k + 2\*i - 1;

for( j= 1;j<=k;j++)

s+= i\*j

}

# Zadaci

1. Programer Luka mnogo voli dinosauruse i za potrebe svojih istraživanja posmatra fiziološke karakteristike *2n* različitih vrsta dinosaurusa (n≤1000000). Uočio je da se dužina repa kod dinosaurusa evoluciono menjala u toku trijasa, jure i krede. Pomozite Luki da nađe najmanju graničnu vrednost *t* tako da od polaznog niza različitih vrsta dinosaurusa može formirati *n* takvih parova, da je zbir dužina repa svakog para manji ili jednak od *t*.

ULAZ: U prvom redu dat je prirodan broj n. U drugom redu dato je 2n realnih brojeva (dužine repa) razdvojenih razmakom.

IZLAZ: Traženi broj t na 4 decimale.

ULAZ IZLAZ

3 13.0000

2 7 6 9 1 12 Vremensko ograničenje: **1 s**, Memorijsko ograničenje: **1000 mb**

2. Programer Srđan mnogo voli da se bavi obradom slika dinosaurusa. Slika1 dinosurusa je dimenzije m\*n i data je kao matrica piksela. Srđan želi da kreira novu sliku2 iste dimenzije tako da se svaki element u **i**-toj vrsti i **j**-toj koloni slike1 zameni minimalnim elementom podmatrice A[i][j] koju obrazuju elementi gornjeg levog ugla matrice (koji nisu desno od kolone **j** i ispod vrste **i**). Sa ulaza se u prvom redu učitavaju brojevi m, n, a potom se u narednih m redova i n kolona učitavaju elementi slike1. Elementi jedne vrste matrice su razdvojeni blankom. Konstruisati algoritam vremenske složenosti do O(mn) koji će na izlazu ispisati sliku2.

3. Programer David igra igricu DinosCraft u kojoj se n (n ≤ 2.000) primeraka argentinosaurusi bori za za plen u močvari sa n primeraka tiranosaurusa. Svaki dinosaurus ima svoj intenzitet udaraca (prirodni broj manji od ili jednak 2.000.000.000) i ne postoje dva dinosaurusa sa jednakim intenzitetom udaraca. David želi da kreira parove za turnir tako da u svakom paru učestvuje po jedan argentinosaurus i po jedan tiranosaurus, a svaki od dinosaurusa se pojavljuje u tačno jednom paru. Turnir se sastoji od *n* dvoboja (u *i*-tom dvoboju (1 ≤ *i* ≤ *n*) učestvuju dinosaurusi *i*-tog para) i pobeđuje dinosaurus koji jače udara. David na početku igrice ima 0 poena. Svaki od *n* tiranosurusa može se opisati sa po dva broja: jedan koji govori koliko David dobija poena ukoliko argentinosaurus pobedi tiranosaurusa i drugi koji govori koliko David gubi poena ukoliko tiranosaurus pobedi argentinosaurusa. Odrediti koliki je maksimalan broj poena koji David može skupiti (može i negativno rešenje). U prvom redu standardnog ulaza nalazi se prirodan broj n. U drugom redu nalazi se n prirodnih brojeva: i-ti od tih brojeva (1 ≤ i ≤ n) predstavlja intenzitet i-tog argentinosaurusa. U trećem redu nalazi se n prirodnih brojeva: i-ti broj u tom redu (1 ≤ i ≤ n) predstavlja intenzitet i-tog tiranosaurusa. U četvrtom redu nalazi se n prirodnih brojeva: i-ti broj (1 ≤ i ≤ n) predstavlja broj poena koji David dobija ukoliko je argentinosaurus poražen. U petom redu nalazi se n prirodnih brojeva: i-ti broj (1 ≤ i ≤ n) predstavlja broj poena koje David gubi ukoliko je tiranosurus izašao kao pobednik dvoboja. Svaki poeni u 4. i 5. redu je manji ili jednak 1.000.

ULAZ IZLAZ

4 488

287934189 130313753 1903712406 1226942245

530539862 815774000 1345413382 989948782

102 214 437 616

287 278 577 339

4. Programer Kosta mora da pomogne špediterskoj kompaniji XYZ u transportu P (0 < P ≤ 109) tona specijalnog močvarnog peska radi potrebe snimanja filma o dinosaurusima. Pesak se mora transportovati u kanisterima kako bi očuvao suvoću. Kompanija XYZ raspolaže kanisterima sa k (1 ≤ k ≤ 100) različitih kapaciteta, a najmanji kapacitet je 1 tona. Kosta zna da na raspolaganju ima potrebno mnogo kanistera svakog od datih kapaciteta, ali kompanija XYZ mora da da izabere što je moguće manje kanistera za transport. Napišite program, koji, za date k, P (1. red ulaza) i kapacitete raspoloživih k kanistera u 2. redu ulaza razdvojenih blankom(jedan od datih kapaciteta je jednak 1), određuje minimalan broj kanistera, koje se koriste za prevoz peska.

ULAZ IZLAZ

999865497 7 1129042

1 267062069 637323855 219276511 404376890 528753603 199747292