Rozdział 1

I egzamin cząstkowy 2014 - potencjalne zadania

1.1.

> Przedstaw ideę algorytmu Boruvki (Sollina).

G - nasz graf G' - nasza odpowiedź, początkowo tylko wierzchołki z G i żadnych krawędzi Wykonujemy kroki:

1. Dla każdego wierzchołka w G znajdź najkrótszą incydentną krawędź i dodaj ją do G'
2. Utwórz nowy graf G', w którym wierzchołkami są spójne składowe w starym G'

Iterujemy do póki nie zostanie jednen wierzchołek. Wszystkie dodane krawędzie do G' utworzą odpowiedź.

1.2.

> Które z poniższych algorytmów mogą działać niepoprawnie dla grafów z ujemnymi wagami krawędzi? Odpowiedź uzasadnij.
>
> a) algorytm Kruskala
>
> b) algorytm Prima
>
> c) algorytm Dijsktry

a) Kruskala - działa dobrze

b) Prima - działa dobrze

c) Dijkstra - Jeżeli gdzieś w grafie występuje cykl o negatywnej wadze, to wszystkie drogi nie mają najkrótszej ścieżki. Algorytm może tego nie wykryć, bo nigdy nie wraca do wierzchołków już rozważonych, a cykl możemy znaleść dopiero pod koniec wykonania, dlatego zwróci jakieś wartości dla wierzchołków, a nie powinien.

1.3.

> Rozwiąż równanie rekurencyjne
> $$T(1) = 1$$
> $$T(2) = 3$$
> $$T(n) = T(n-2) + 2n - 1 \qquad \text{dla } n > 2$$

$$T(1) = 1$$
$$T(2) = 3$$
$$T(n) = T(n-2) + 2n - 1 = T(n-2) + n + (n-1) \qquad \text{dla } n > 2$$

$$\begin{aligned} T(n) &= n + (n-1) + T(n-2) = \\ &= n + (n-1) + (n-2) + (n-3) + T(n-4) = \\ &= n + (n-1) + (n-2) + (n-3) + \cdots + 4 + 3 + 2 + 1 = \\ &= \frac{n(n+1)}{2} \end{aligned}$$

1.4.

> Który z poniższych algorytmów sortowania może w najgorszym przypadku wykonać $\Omega(n^2)$ porównań:
>
> a) quicksort
>
> b) mergesort
>
> c) insertsort
>
> Przypomnienie: $\Omega(n^2)$ oznacza nie mniej niż cn^2 dla pewnej stałej $c > 0$.

a) quicksort - jeżeli znajdowanie pivota jest źle zrobione, tzn. odcina stałą liczbę elementów przy każdym partition, np. stosunek $1 : (n-1)$, to wtedy złożoność to $n * O(n) = O(n^2)$

b) mergesort - worst case $O(n \log n)$

c) insertsort - posortowany lub odwrotnie posortowany ciag bedzie mieć $O(n^2)$

1.5.

> Napisz procedurę partition (nie musi być to wersja z wykładu, ale musi być efektywna).

Pivot jest w A[p], funkcja zwaraca granicę podziału

```
Part(A[1..n], p, r)
  x <- A[p]
  i <- p-1
  j <- r+1

   while(i<j) do
     repeat(j--) until A[j] <= x
     repeat(i++) until A[i] >= x

     if(i<j) swap(A[i], A[i])
     else return j
```

1.6.

> Przedstaw strategię zachłanną algorytmu aproksymacyjnego dla problemu Set Cover o współczynniku aproksymacji H_n.

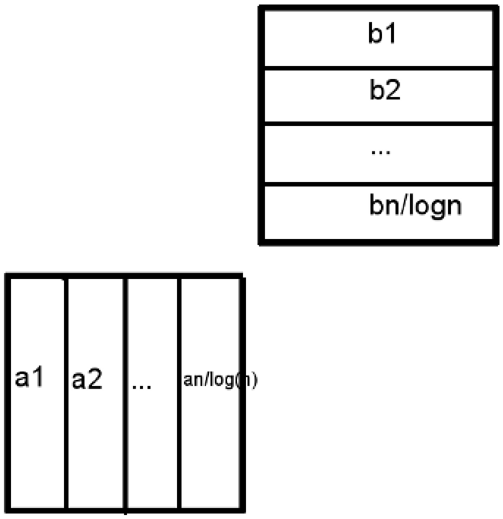
Problem: Wybrać najmniejszy taki podzbiór z rodziny zbiorów, aby wszystkie pojedyncze elementy z całej rodziny znalazły sie w tym podzbiorze. Algorytm aproksymacyjny: W każdym kroku odrzucamy zbiór, który nie pokrywa jak nawiekszej części elementów.

```
U - uniwersum do pokrycia
R - wejsciowa rodzina zbiorow
W - wynikowa rodzina zbiorow na poczatu pusta

while U jest niepusty
    wybierz takie Z z R ze Z przekroj U jest maksymalne
    dodaj Z do rodziny wynikowej W
    U \= Z
    usun Z z R
```

1.7.

> W algorytmie czterech Rosjan obliczane są iloczyny macierzy o rozmiarze $nx\log n$ i macierzy o rozmiarze $\log nxn$ Ile takich iloczynów jest obliczanych?



Liczymy iloczyny macierzy dla każdego $a_i \cdot b_i$. Takich iloczynów jest oczywiscie $\frac{n}{\log(n)}$, a każdy w wyniku daje jedną macierz kwadratową. Jeśli zsumujemy te macierze to otrzymamy iloczyn, o który nam chodzi.

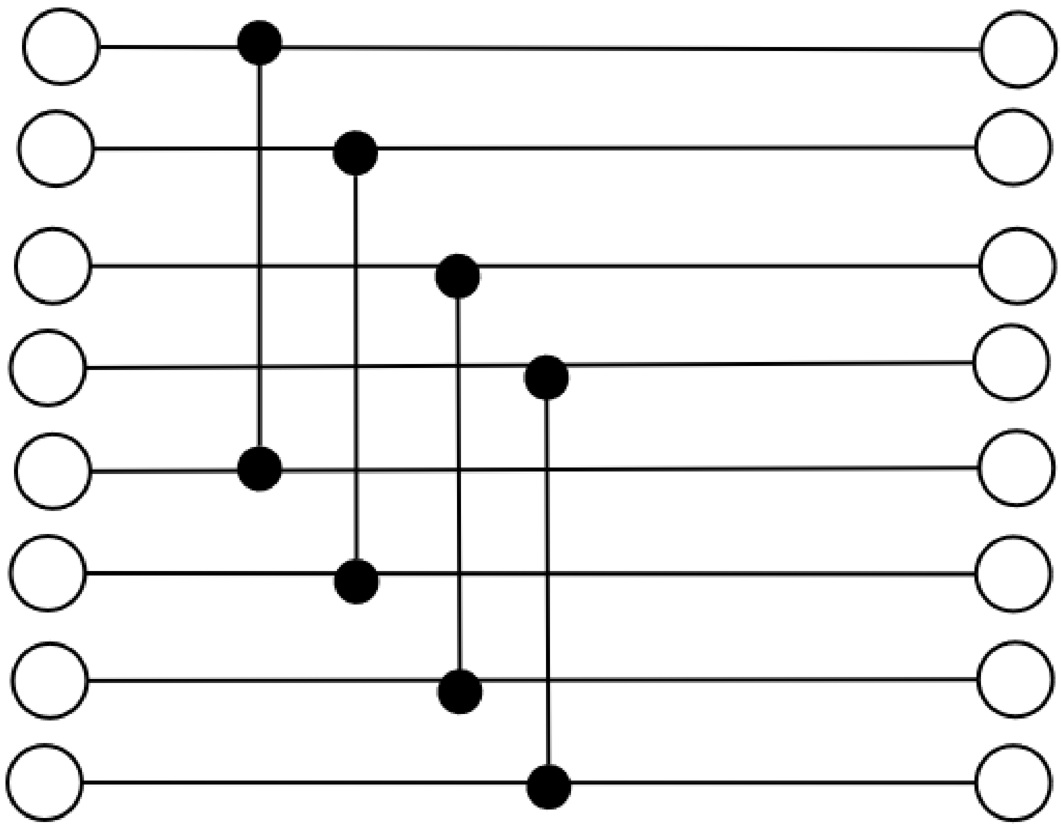
1.8.

Opisz, w jaki sposób DFS może być zastosowane do znalezienia cyklu Eulera w grafie.

Graf przechodzimy przy pomocy rekurencyjnej procedury DFS. Przebyte krawędzie usuwamy, a wierzchołki po zakończeniu przetwarzania umieszczamy na początku kolejki. Jeśli graf posiada cykl Eulera, to po zakończeniu algorytmu w kolejce znajdą się kolejne wierzchołki tego cyklu.

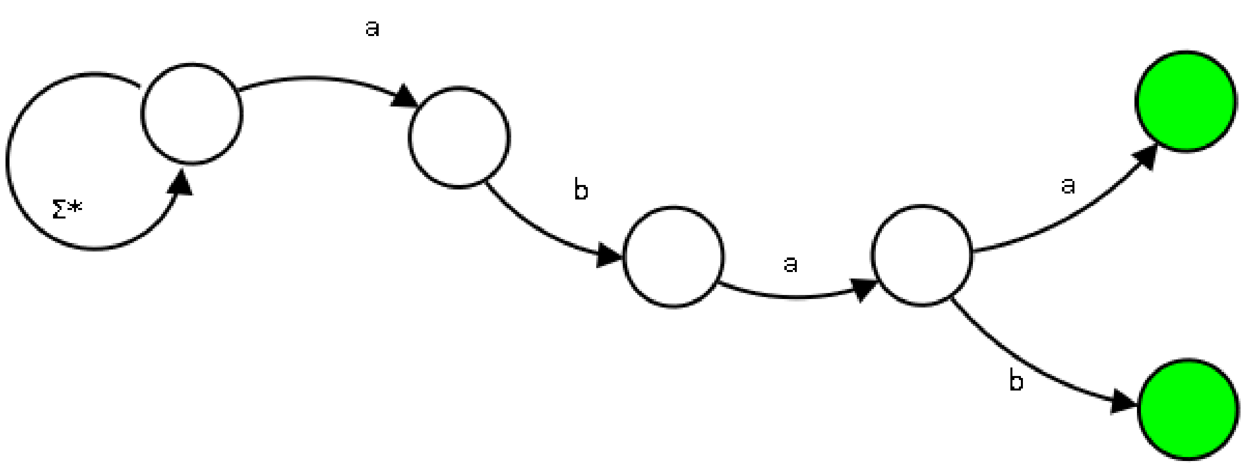
1.9. ??? tego chyba nie było jeszcze

Narysuj sieć półczyszczacą o ośmiu wejściach.



1.10. ??? tego chyba nie było jeszcze

Narysuj automat rozpoznający *abaa* i *abab*.



1.11.

W jaki sposób problem mnożenia macierzy może być wykorzystany do rozwiązania problemu najkrótszych ścieżek w grafie?

Mamy grafy A i B w postaci macierzy adjacencji.
Możemy korzystać z algebry, w której:
'+' to działanie min
'·' to dodawanie
Wtedy jedno mnożenie macierzy, to jeden krok relaksacji najkrótszych ścieżek, musimy wykonać n takich relaksacji. Złożoność $O(n^4)$. Można poprawić złożoność. Nie da się zastosować algorytmu Strassena, bo nie mamy działania odwrotnego do min. Można za to skorzystać z szybkiego potęgowania, co zmniejsza nam koszt do $O(n^3 \log n)$.

1.12.

Opisz, w jaki spósb obliczenie wielomianu n-tego stopnia w n-tych pierwiastkach jedności jest redukowalne do obliczenia dwóch wielomianów stopnia $n/2$ w $(n/2)$-tych pierwiastkach jedności.

Mamy wielomian $A(x)$ stopnia n, reprezentowany przez współczynniki $a_0, a_1, \cdots, a_n$.
Tworzymy 2 nowe wielomiany:
$A^{[0]}(x) = a_0 + a_2x + a_4x^2 + ...a_{n-2}x^{n/2-1} + a_nx^{n/2}$
$A^{[1]}(x) = a_1 + a_3x + a_5x^2 + ...a_{n-1}x^{n/2-1}$

$$A(x) = A^{[0]}(x^2) + xA^{[1]}(x^2)$$

Rozdział 2

nierozwiązane

2.1.

Opisz algrotym mnożenia długich liczb (n bitowych)?

2.2.

Jak posortwoać n liczb z przedziału $[1, n^2]$ w czasie liniowym?

2.3.

Ile warstw musi mieć siec przełączników, żeby dalo się uzyskać wszystkei przesunięcia cykliczne ciągu n-elemntowego?

Bibliografia