Algoritmi e Laboratorio
a.a. 2004/2005

Prova scritta del 8 Luglio 2005 (FOGLIO 1 DI 2, con 2 facciate)

COGNOME ………………………. NOME ………..………………

Matr. n. …………………..

1. (PUNTI 2 + 2) Completare i seguenti due algoritmi con gli <u>invarianti di ciclo</u> che permettono di dimostrarne la correttezza e le asserzioni finali.

<u>ALGORITMO 1</u>
{A.I.: $n \geq 0$ & A[0..n] contiene $a_0 \dots a_n$ }
result ← A[0]
i ← 1

{I.C. : ……………………………………………………………………………………}
while i ≤ n **do**
    result ← result *A[i]
    i ← i + 1

return result
{A.F.: result = ………………………………………………………………………… }

<u>ALGORITMO 2</u>
{A.I.: $n \geq 0$ & A[0..n] contiene $a_0 \dots a_n$ }
result ← A[n]
i ← n − 1

{I.C.: ……………………………………………………………………………………}
while i ≥ 0 **do**
    result ← result * A[i]
    i ← i − 1

return result
{A.F.: result = …………………………………………………………………………}

2. (PUNTI 1+1+2)

a) Fornire un lower bound al numero di confronti richiesti per ordinare n oggetti nel modello basato su confronti.
b) Fornire la definizione di "algoritmo di ordinamento stabile".
c) Indicare le complessita' in spazio (vettore che memorizza l'input escluso) e in tempo dell'algoritmo ottimo per ordinare un array di *n* elementi contenente una permutazione degli *n* interi compresi tra *1* e *n* .

3. (PUNTI 2+1)
Risolvere la seguente equazione di ricorrenza:

T(1) = 2   per n<=1
T(n) = (2/3)*T(⌊n/3⌋) + (1/4)*T(⌊n/3⌋) + (1/12)*T(⌊n/3⌋) + n^2   per n >1

- mediante il metodo iterativo
- mediante applicazione del teorema principale

4. (PUNTI 1+4)

a) Scrivere lo "schema generale" dell'algoritmo Divide-et-impera.

b) Applicarlo per trovare un algoritmo di soluzione per il seguente problema:

Data un array A di n elementi (n potenza di 4), restituire il valore:

A[1] + A[5] ^ 5 + A[9] ^ 9 + A[13] ^ 13 + … + A[n-3] ^(n-3)

5. (PUNTI 1 + 2 + 1)

a) Fornire l'algoritmo *Greedy-Activity-Selector* .

b) Applicare l'algoritmo *Greedy-Activity-Selector* al seguente insieme di attività, di cui sono forniti gli intervalli di attività [s, f).

	s	**f**
A	**7**	**13**
B	**3**	**8**
C	**6**	**8**
D	**7**	**9**
E	**7**	**10**
F	**3**	**9**
G	**5**	**7**
H	**4**	**5**

c) L'insieme massimale di attività mutuamente compatibili trovato è l'unico per l'insieme di attività dato? (MOTIVARE LA RISPOSTA.)

6. (PUNTI 1+2) Sia dato il grafo non orientato rappresentato dalla seguente lista di adiacenza:

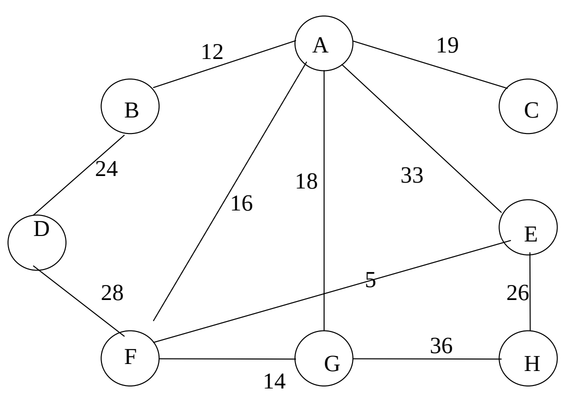
A	B C E F G
B	A D
C	A
D	B F
E	A F H
F	A D E G
G	A F H
H	E G

a) Se ne effettui la visita in profondità, considerando H come vertice sorgente, e si indichino per ogni vertice i tempi di inizio e fine visita.

b) Si disegni il grafo, etichettando ogni suo arco come T (dell'albero), B (all'indietro), F (in avanti), C (di attraversamento), avendo sempre H come sorgente *(attenti al"tranello").*

7. (PUNTI 5 + 1)

a) Si determini, applicando l'algoritmo di Prim a partire dal vertice F, un minimum spanning tree del seguente grafo non orientato, connesso, e pesato.



Algoritmi e Laboratorio
a.a. 2004/2005

Prova scritta del 8 Luglio 2005 (FOGLIO 2 DI 2, con 1 facciata)

COGNOME ………………………. NOME ………..………………

Matr. n. …………………..

Per svolgere correttamente l'esercizio occorre:

- compilare le seguenti tabelle (la riga 0 è già compilata), e
- disegnare (un disegno per ogni riga delle tabelle) l'albero mantenuto dall'algoritmo (contenente SIA gli archi che fanno parte della soluzione CHE gli archi candidati) al termine di ogni iterazione (cioe' DOPO che sono state aggiornate le appetibilita' dei vertici in coda).

La prima tabella indica, per ogni iterazione del ciclo esterno, per ogni vertice v che non appartiene alla soluzione (l'albero definitivo), il peso di un arco di peso minimo che collega tale vertice ad un vertice della soluzione (∞ **se tale arco non e' ancora stato trovato)**. Il valore di k non deve essere più riportato quando il vertice e' ormai parte della soluzione (si metta il simbolo -).

La seconda tabella indica, per ogni iterazione del ciclo esterno, l'insieme dei vertici v che sono entrati a far parte della soluzione (per i quali k[v] è il peso dell'arco incluso).

La riga 0 corrisponde al temine dell'inizializzazione (prima di entrare nel ciclo).

Quando nella coda con priorità ci sono vertici con lo stesso valore minimo di k si scelga quello che viene prima secondo l'ordine alfabetico.

k	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**
0	∞	∞	∞	∞	∞	**0**	∞	∞
1								
2								
3								
4								
5								
6								
7								
8								

	Insieme dei vertici t inclusi nella soluzione
0	**{ }**
1	
2	
3	
4	
5	
6	
7	
8	

b) L'albero ottenuto e' l'unico albero di copertura minimo per il grafo dato? (MOTIVARE LA RISPOSTA.)

8. (PUNTI 4) Dati un intero *k*, un grafo *G* non orientato e connesso con n vertici, e un vertice *s* di G. Si scriva un algoritmo che, per mezzo di una visita del grafo *G* calcoli:

- il numero dei vertici che <u>nel grafo</u> si trovano a distanza: < *k*,
- il numero dei vertici che <u>nel grafo</u> si trovano a distanza non uguale a *k*, e
- il numero dei vertici che <u>nel grafo</u> si trovano a distanza > *k*,

dal vertice *s*.