

# PROGETTO ALGRAPH

---

Angelo Di Iorio

Università di Bologna

# Progetto ALGRAPH

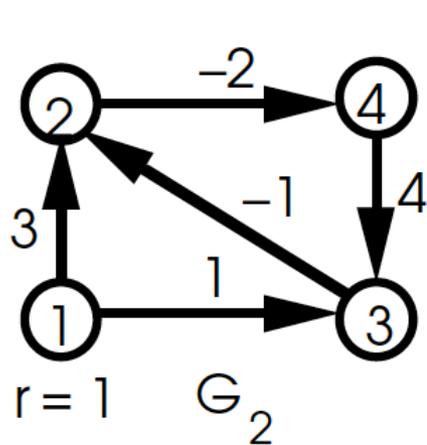
- ALGRAPH (ALGA for GRAPHS) è un software Java in grado di simulare e documentare l'esecuzione di un algoritmo su grafi
- Applicazione interattiva basata su Java FX
- Obiettivo: realizzare un software utile per spiegare uno tra gli algoritmi visti a lezione su grafi pesati
  - e usare il software il giorno della discussione per spiegare l'algoritmo

# Algoritmi

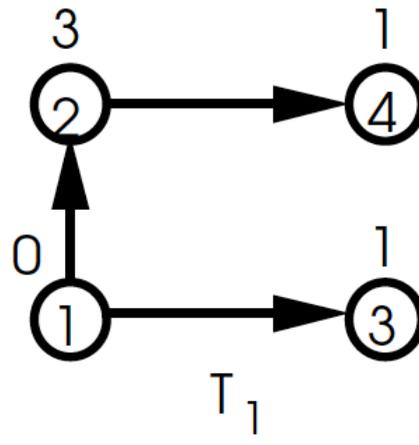
- Il progetto si svolge in gruppo (2-3 persone) e ogni gruppo sceglie un algoritmo da un elenco predefinito
- Algoritmi su grafi pesati:
  - *Albero dei cammini minimi*
    1. *Dijkstra*
    2. *Johnson*
    3. *Bellmann-Ford-Moore*
  - *Minimo albero di copertura*
    4. *Kruskal*
    5. *Prim*
- Ogni algoritmo può essere scelto da **5 gruppi al massimo** (se non bastano rilasceremo il vincolo)

# Albero dei cammini minimi

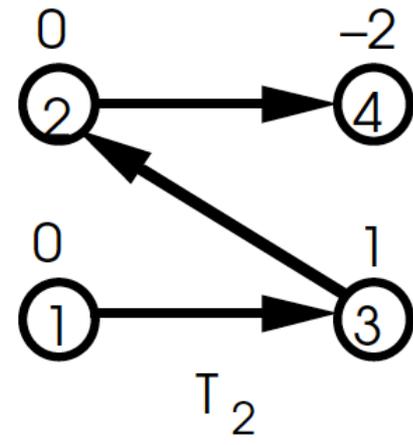
- A lezione vedrete alcuni algoritmi in dettaglio



b)



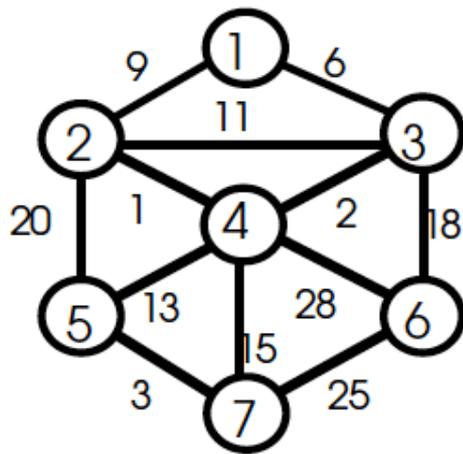
c)



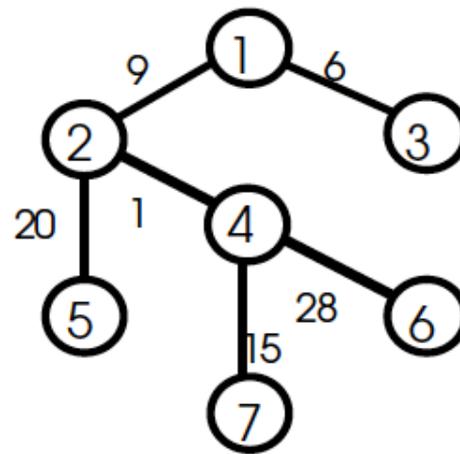
d)

# Albero di copertura minimo

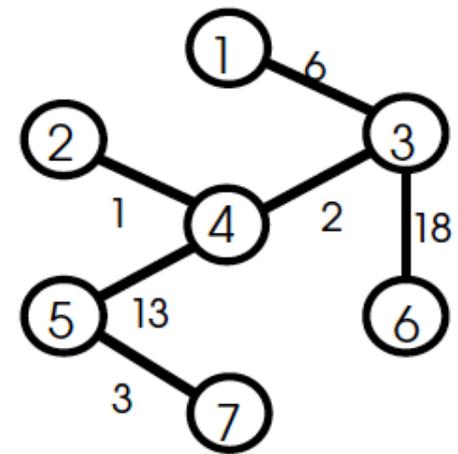
- A lezione vedrete alcuni algoritmi in dettaglio



a)



b)



c)

# Regole: gruppi e voto

- Il progetto si svolge in gruppo di 2 o 3 persone
- **NON sono ammessi gruppi di più di 3 persone**
- Il progetto si consegna una volta sola e il voto è valido per l'intero anno accademico (e per gli anni successivi se le regole non cambiano)
- E' obbligatorio consegnare il progetto prima dell'esame scritto
- Il progetto "fa media" (pesata) con il voto dello scritto:  $\frac{1}{4}$  del voto complessivo
- Il voto non dipende dall'algoritmo scelto ma dallo svolgimento del progetto

# Regole: consegna

- Il progetto si discute di persona (tutti i membri del gruppo insieme) e si consegna due giorni prima della discussione
  - mail a: [angelo.diiorio@unibo.it](mailto:angelo.diiorio@unibo.it)
  - link a spazio condiviso (Google Drive, Dropbox o altri spazi)
- La consegna finale deve includere:
  - Sorgenti Java
  - Eseguibile, Runnable .jar
  - Javadoc
  - Relazione (poche pagine) su struttura dell'applicazione e README

# Quando consegnare?

- Orientativamente 2 settimane prima di ogni appello scritto
- A giugno/luglio:
  - 23 maggio (scritto: 6 giugno)
  - 14 giugno (scritto: 26 giugno)
  - 4 luglio (scritto: 18 luglio)
- Date e aule da confermare (ma cambieranno di poco)
- Saranno pubblicate su AlmaEsami e sul sito del laboratorio
- **Sufficiente iscrivere un membro del gruppo**
- Se ci sono molte richieste possiamo fissare un'ulteriore data straordinaria prima o dopo queste

# ALGRAPH: requisiti

- Siete liberi di scegliere le informazioni che ritenete utili per spiegare l'algoritmo ma sono richiesti:
  - descrizione testuale e pseudocodice dell'algoritmo
  - campi/bottoni per inserire dati in input e interagire con l'applicazione
  - strutture dati utilizzate e come queste cambiano durante l'esecuzione
- ALGRAPH permette di generare il grafo di input casualmente, di importarlo da un file esterno e di modificare il grafo corrente
  - Informazioni e operazioni dipendono dal tipo grafo usato
- ALGRAPH permette di eseguire l'algoritmo per passi o guardare il risultato finale
- ALGRAPH gestisce correttamente gli errori e li notifica all'utente
  - input non corretto, errori di selezione, etc.

# Dati in input

- All'apertura ALGRAPH permette all'utente di creare il grafo su cui lanciare l'algoritmo:
  - 1) grafo generato casualmente
    - l'utente deve poter scegliere il numero di nodi
  - 2) dati caricati da un file
    - strutturato secondo regole stabilite dall'applicazione
    - ALGRAPH verifica se il file è valido ed è possibile caricare i dati; in caso contrario notifica all'utente la presenza di errori
- L'utente deve poter modificare interattivamente il grafo in input
  - aggiungere o eliminare un nodo
  - modificare il peso di un arco
  - modificare la direzione di un arco
- ALGRAPH permette di esportare il grafo corrente in un file

## Dati in input (2)

- I dati dipendono dal tipo di grafo usato
- I pesi degli archi sono valori interi, eventualmente negativi
- I nodi del grafo possono essere etichettati con numeri interi, lettere o altre informazioni purchè siano visualizzate correttamente
- ALGRAPH verifica che i dati inseriti dall'utente siano corretti
- E' possibile imporre vincoli sui grafi in input (es. dimensione) ma deve essere correttamente notificato all'utente

# Esecuzione

- ALGRAPH permette all'utente di selezionare la modalità di esecuzione e animazione dell'algoritmo:
  - *Step-by-step*: l'utente clicca su un bottone per avanzare di un "passo" nell'esecuzione
  - *Final Result*: l'utente clicca su un bottone per vedere il risultato finale
- Ad ogni passo, ALGRAPH mostra le strutture dati utilizzate e come queste cambiano durante l'esecuzione
  - Siete liberi di scegliere messaggi di spiegazione, pop-up, animazioni o altri hints grafici

# Come scegliere e prenotarsi

- Compilare form: <http://diiorio.nws.cs.unibo.it/asd1718/progetto/>
- Non è possibile cambiare gruppo o algoritmo tramite il form, contattare direttamente il docente via mail

Algoritmi e Strutture Dati - Progetto ALGRAPH Info

---

Gruppo

Nome Gruppo:

---

|                      |                      |          |                      |             |                      |
|----------------------|----------------------|----------|----------------------|-------------|----------------------|
| Componente 1 - Nome: | <input type="text"/> | Cognome: | <input type="text"/> | Mail Unibo: | <input type="text"/> |
| Componente 2 - Nome: | <input type="text"/> | Cognome: | <input type="text"/> | Mail Unibo: | <input type="text"/> |
| Componente 3 - Nome: | <input type="text"/> | Cognome: | <input type="text"/> | Mail Unibo: | <input type="text"/> |

---

Algoritmo

**Attenzione:** il form non richiede conferma prima di spedire i dati. Cliccare su INVIA con la scelta definitiva.

Dijkstra  
 Johnson  
 Bellmann-Ford  
 Kruskal  
 Prim

# Cosa ci aspettiamo dal progetto?

- Il progetto deve:
  - Implementare correttamente l'algoritmo
  - Mostrare chiaramente la sua esecuzione
  - Gestire gli errori
  - Gestire le diverse modalità di input e la modifica del grafo
  - Avere un layout gradevole
- **Soprattutto: deve essere frutto del vostro lavoro, dovete essere in grado di spiegarlo e modificarlo on-the-fly il giorno della consegna!**

# Domande?

- Possiamo usare Java FXML?

*Si*

- Possiamo usare Java SWING?

*Si (ma sconsigliato)*

- Altre domande?