

## Corso di Laboratorio di Programmazione

# Sistema di approvvigionamento

### 2° assegnamento – progetto 1

## Introduzione

L'azienda per cui lavorate produce elettrodomestici, e ha deciso di passare a una gestione informatizzata dell'acquisto dei componenti necessari a produrli. Per ogni elettrodomestico che deve essere prodotto, esiste un elenco di componenti che è necessario impiegare. Ciascun componente ha un dato tempo di arrivo che si misura in mesi, ed è necessario avere a disposizione tutti i componenti per procedere alla produzione di un elettrodomestico.

Vi si chiede di creare il software che gestisce gli acquisti dei vari componenti e la messa in produzione degli elettrodomestici, in modo da evadere nel più breve tempo possibile gli ordini in arrivo.

Un esempio: supponete di produrre un elettrodomestico che ha bisogno dei componenti A, B e C, con tempi di arrivo di 2, 4 e 3 mesi rispettivamente. Per produrlo dovreste ordinare al mese 1 i componenti di tipo B, il mese successivo (2) i componenti di tipo C, e quello ancora successivo (3) i componenti di tipo A, e la produzione può partire dal mese 5. La produzione di un lotto di elettrodomestici dura un mese (indipendentemente dal numero di pezzi prodotti).

Il sistema da realizzare è descritto in due passi. Allo step 1 è presentato il sistema in versione semplificata, mentre lo step 2 aggiunge ulteriori elementi. Il progetto è considerato completo solo se sono svolti entrambi gli step.

## Step 1: produzione

Il sistema da realizzare deve essere in grado di gestire la produzione di un numero qualsiasi di elettrodomestici. Un singolo componente può essere presente in più di una tipologia di elettrodomestico. Il vostro software acquisisce informazioni sugli elettrodomestici da produrre accedendo al file **models.dat**, che contiene un elenco di nomi di file, ciascuno dei quali contiene l'elenco dei componenti necessari per produrre un determinato elettrodomestico. Per esempio, il file **models.dat** può contenere i seguenti nomi di file: **model1.dat** e **model2.dat**. Ciascuno di questi file contiene una prima riga con le informazioni riguardo all'elettrodomestico da produrre:

```
[model_id] [model_name]
```

segue quindi l'elenco dei componenti in questo formato:

```
[component_id] [component_name] [quantity_needed]
```

che contiene, nell'ordine, l'id del componente (un numero), un nome associato, e la quantità di componenti necessaria per la produzione di un modello di elettrodomestico. La ridondanza riscontrabile tra id e nome (sia del modello che del componente) è necessaria per poter comunicare facilmente con l'utente del sistema.

Il sistema trova le informazioni sul tempo di arrivo dei componenti nel file **components\_info.dat**, che contiene righe con il seguente formato:

```
[component_id] [component_name] [delivery_time (months)]
```

Le informazioni sugli ordini di elettrodomestici in arrivo si trovano nel file **orders.dat**, che ha il seguente formato:

[time\_stamp] [model\_id] [quantity]

Tutti i file .dat sono dei normali file di testo; l'estensione .dat serve solo a sottolineare che sono file contenenti dati, e hanno perciò un particolare formato.

Nel progetto è possibile supporre che i file abbiano il formato corretto: non è necessario effettuare verifiche di formato.

Il sistema deve contenere una funzione che stampa lo stato corrente di:

- acquisti effettuati (componenti acquistati ma non ancora arrivati);
- magazzino (componenti ordinati e arrivati, ma non ancora utilizzati per la produzione);
- ordini evasi (elettrodomestici prodotti e consegnati).

Tale funzione deve essere chiamata ogni volta che un ordine è evaso.

## Step 2: produzione e gestione economica

In questo secondo step si tiene conto della disponibilità di cassa dell'azienda produttrice. La cassa a disposizione (in euro) è fornita al sistema come primo elemento del file orders.dat. I file di descrizione di ogni elettrodomestico hanno alla prima riga un campo aggiuntivo con il prezzo di vendita del modello, mentre il file components\_info.dat ha tre campi aggiuntivi: il prezzo per un ordine di 1-10 componenti, quello per un ordine di 11-50 componenti, e quello per l'ordine di 51+ componenti. In questo secondo step è possibile effettuare un ordine di componenti solo se la cassa riesce a coprirlo (non è possibile andare sotto zero con la cassa). Il sistema deve organizzare la produzione di conseguenza. In base agli ordini ricevuti e alla cassa a disposizione, potrebbe non essere possibile produrre tutti gli elettrodomestici ordinati in un unico lotto di produzione.

## Indicazioni per lo svolgimento

Il progetto deve essere suddiviso in più file sorgente. **Ogni file deve essere scritto da un solo studente.** È tuttavia possibile controllare che il codice dei propri compagni di gruppo funzioni correttamente. Un errore in un file che inficia il funzionamento del progetto potrebbe causare una penalizzazione della valutazione dell'intero gruppo. **Il nome dell'autore deve essere indicato in un commento a inizio file.** Ovviamente, è necessario e positivo che si discuta all'interno di ciascun gruppo su come realizzare il codice, ma poi ogni studente è responsabile della gestione del codice che deve scrivere.

Non sono volutamente presenti vincoli o suggerimenti su quali classi creare per rappresentare il problema. Siete liberi di progettare come preferite. Saranno valutati:

- Chiarezza e correttezza del codice;
- Corretta gestione della memoria;
- Efficienza del codice e della soluzione trovata;
- Utilizzo di strumenti appropriati.

Il compito richiede la gestione dei file. Questa è un'estensione della gestione degli stream che abbiamo visto a lezione, e può essere facilmente appresa sfruttando il libro o la numerosa documentazione online.

La pianificazione della produzione è un problema potenzialmente molto complesso. Tuttavia, lo scopo di questo progetto è la creazione di software, quindi eventuali soluzioni sub-ottime non saranno penalizzate. In altre parole: il focus è sulla produzione di software più che sulla ottimalità dell'algoritmo di ottimizzazione.

Il software deve essere basato unicamente sulla libreria standard del C++.

## Consegna

Il compito deve essere consegnato su Moodle, caricando un archivio che include una directory contenente:

- Il codice sorgente;
- un file models.dat e i file di descrizione di singoli elettrodomestici che sono elencati in models.dat (almeno 4 tipi di elettrodomestici);
- un file components\_info.dat con almeno 10 componenti;
- file di input orders.dat (almeno quattro) che avete usato per provare il vostro software, scritti secondo i criteri descritti nel paragrafo "input";
- un file di testo contenente:
  - riga di comando usata per compilare, con path relativi (cioè non devono comparire "/home/mio\_nome/dir/progetto");
  - qualche esempio di output di esecuzione.

L'archivio non deve contenere l'eseguibile, perché il sorgente sarà compilato in fase di correzione. Vi informo che il software sarà compilato con le opzioni di ottimizzazione (-O2), per rendere più evidenti eventuali memory leak. Siete invitati a fare altrettanto per i vostri test.

Suggerisco di consegnare anche compiti non completi. È tuttavia necessario che il software consegnato sia compilabile ed eseguibile.