Manuale per il docente:

Linee guida per l’attività “Registro Esami”:

1. Introduzione:

Lo scopo di questa attività è quello di abituare gli studenti ad essere in grado di trasformare il codice sorgente di un dato linguaggio, in un altro. Nella fattispecie, in questo caso, abbiamo il linguaggio C ed il linguaggio Java. L’obiettivo è quindi quello di saper convertire il codice da Java in C e viceversa. Considerando il fatto che in parallelo a questo corso si sta svolgendo anche il corso di Java avanzato “Object Oriented Software Design”, è opportuno e interessante fare dei paralleli immediati tra questi due linguaggi, anche per sottolineare differenze e similitudini tra i vari schemi di programmazione.

1. Prerequisiti e Modelling:

Per quanto riguarda le conoscenze preliminari per la parte in C, è opportuno che gli studenti abbiano bene in mente il concetto di struttura e di puntatore. Questi due concetti sono infatti necessari per fare dei paralleli immediati con il linguaggio Java. Per quanto riguarda le conoscenze del linguaggio Java, è necessaria la conoscenza del concetto di classe ed istanza di una classe, e saper accedere ai metodi e le variabili di queste (è quindi necessaria anche la conoscenza dei modificatori di accesso, e le varie differenze tra questi). Ad esempio, un buon esercizio preliminare può essere quello di creare una classe “Persona”, composta di tutti i suoi vari dati anagrafici. Una volta creati 5 oggetti di tipo Persona, si chiede di cambiare il nome ad almeno uno di questi, ed effettuare un ordinamento in base all’età. Il passo successivo è proporre la stessa soluzione in linguaggio C.

1. Concetti richiamati:

In questa attività vengono messe in risalto due concetti fondamentali per il C e Java: le strutture e le classi, rispettivamente. Dal corso preliminare di Java, lo studente dovrebbe avere bene in mente come è strutturata una classe: una classeè la definizione di un tipo di oggetto, e specifica il nome e il tipo delle variabili di istanza degli oggetti. Le strutture ricordano un po' la forma delle classi, ma sono completamente diverse nella loro definizione. Esplicitare queste differenze.

1. Attività e Coaching:

Di seguito ecco una attività da proporre agli studenti:

Si richiede di creare, da riga di comando, un registro per gli esami. Nello specifico, sono dati i seguenti comandi:

0 – Termina inserimento e stampa verbale;

1 – Inserisci nuovo esito;

2 – Annulla esito;

Deve essere possibile inserire i seguenti dati per uno studente: nome, matricola, esito, numero esercizi svolti. Una volta terminata la fase di inserimento, il programma deve stampare a video i primi 3 studenti con voto maggiore, ordinati in ordine decrescente.

Proporre lo stesso programma in linguaggio C.

Coaching: Come già detto, è indispensabile una buona conoscenza delle classi in Java e della loro strutturazione. Un possibile errore può essere quello di creare un oggetto di tipo Studente senza usare il costrutto new, e quindi senza far riferimento all’indirizzo di memoria dell’oggetto creato. Quindi, ad esempio,

Persona persona;

//ERRORE: Non ho istanziato correttamente l’oggetto

System.out.println (persona.getNome());

Invece:

Persona persona = new Persona();

System.out.println(persona.getNome());

Ora ho assegnato un indirizzo di memoria alla variabile persona, l’operatore **new** predispone le variabili di istanza all’interno dell’oggetto creato.

Non solo: questa esercitazione rappresenta un’ottima opportunità per mettere sotto la lente di ingrandimento due concetti come le classi in Java e le strutture in C. A primo impatto, queste sembrano essere molto simili dalla loro struttura. Tuttavia, se sembrano simili dalla loro struttura, non si può dire lo stesso nella loro definizione. Le classi in Java e le strutture in C sono, infatti, due strutture completamente diverse nella loro definizione (e di conseguenza anche nella loro implementazione).