****

# ISW – Gruppo 01 – TeamSoftwareRevolution

Metrica

# Sommario

1. Introduzione
2. Complessità dei prodotti software
   1. Linguaggi imperativi
   2. Linguaggi Object-Oriented
3. Metriche
   1. Tempi di risposta
   2. Modularità
   3. Velocità di completamento
4. Errori
   1. Definizione
   2. Classificazione
   3. Testing
5. Disciplime e fondamenti scientifici
6. Conclusioni

# Introduzione

Il presente elaborato tratta alcuni aspetti fondamentali dell’ingegneria del software.

Inoltre si è descritta una disciplina, che come l’ISW, sta cercando di trovare un proprio fondamento scientifico.

Attraverso un costruttivo lavoro di gruppo, partendo dal lavoro e dalle conoscenze individuali, sono stati argomentati gli aspetti sopra riportati.

Nei paragrafi seguenti, vengono trattati temi quali la complessità dei prodotti software, le metriche di valutazione e tipologie di errori, evidenziando difetti e mancanze.

# Complessità dei prodotti software

Un prodotto software è rappresentato dall'insieme di programmi, procedure di realizzazione, documentazione di sistema, i dati di configurazione e il manuale di riferimento.

Un prodotto software può essere di due distinte tipologie:

* Generico: destinato all’uso di massa (genarl porpose)
* Specifico: realizzato per le esigenze di uno specifico utente/azienda (specific porpose)

La realizzazione di un prodotto software è un processo lungo il quale richiedere la collaborazione di diverse figure professionali specializzate (sviluppatori, team leader, analisti ecc) i quali, insieme agli investitori, utenti ed altre figure interessate ,rappresentano gli stakeholders del prodotto in questione.

Il prodotto software nella generalità dei casi risulta molto complesso, pertanto necessita di una suddivisione in moduli, ciascuno dei quali viene implementato e curato da un diverso team di sviluppo.

Si nota che un possibile criterio per valutare la complessità di un prodotto software è rappresentato appunto dal numero di moduli e sotto-moduli che lo compongono, pertanto la complessità cresce all’aumentare dei moduli.

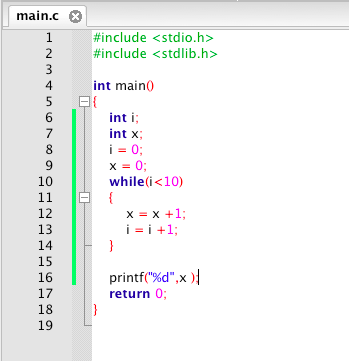
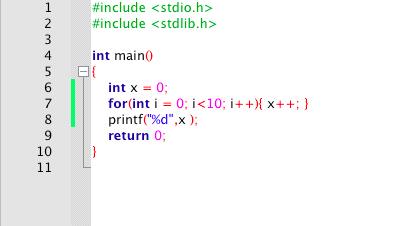
La dimensione di un programma, inteso come numero di righe di codice, pur avendo un impatto visivo notevole, non rappresenta sicuramente un indice di valutazione della complessità.

Due codici differenziati dal numero di righe infatti, a parità di dati inseriti in input, possono fornire entrambe la stessa soluzione.

A conferma di questa tesi, esistono diversi stili di programmazione che si differenziano per la compattezza del codice.

Diversi programmatori preferiscono un codice più “esteso”, e di conseguenza più leggibile, piuttosto che un codice più compatto e di difficile interpretazione.

Di seguente figura fornisce un esempio di due codici con risolvono lo stesso problema con scelte sintattiche e stilistiche differenti:



1.a 1.b