Editoria Elettronica

Vincenzo Gervasi
Dipartimento di Informatica
Università di Pisa

email: gervasi@di.unipi.it
www: http://www.di.unipi.it/~gervasi

Logistica del corso

- Orario delle lezioni:
  - Prima parte (Ottobre)
    - **Lunedì 16:15-17:45 aula A1**
    - **Giovedì 12:00-13:30 (aula ?)**
  - Seconda parte (Novembre, Dicembre)
    - **Lunedì 16:15-17:45 aula A1**
    - **Giovedì 12:00-13:30 laboratorio (I-Lab?)**

Logistica del corso

- Ricevimento studenti:
  - **Martedì 18:00 studio 333** (Dip. Inf.)
  - **altro giorno/orario?**
- Per casi particolari:
  - su appuntamento, spedire email a gervasi@di.unipi.it
  - durante le pause delle lezioni

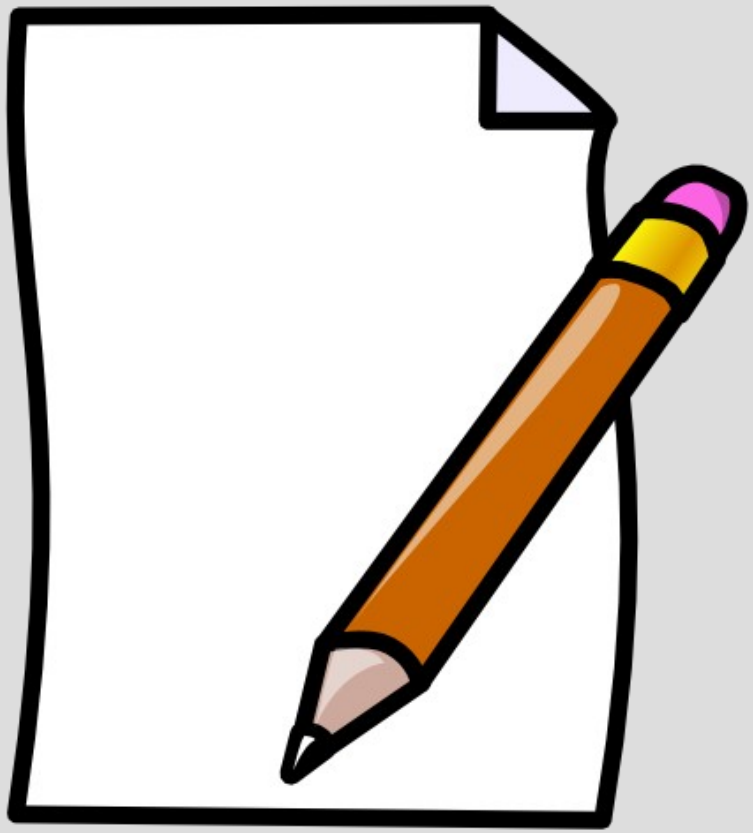
Logistica del corso

- Il corso (5 CFU) viene offerto a:
  - studenti del Corso di Laurea (triennale) in Informatica Umanistica
  - studenti del Corso di Laurea Specialistica in Informatica Umanistica (percorso “Editoria Elettronica”)
  - studenti di altri corsi di laurea (come esame facoltativo)

Logistica del corso

- Valutazione
  - prove in itinere (esercitazioni)
  - un elaborato finale (grafico o programma)
  - una prova orale (teorica)
- Materiale didattico
  - dispense del docente (lucidi, senza animazioni!)
  - altro materiale, anche online, verrà indicato di volta in volta durante il corso

Obiettivi del corso



- Formare **figure professionali** capaci di operare in **ambienti digitali** per la **gestione** e la **pubblicazione** di **informazioni strutturate** e **non strutturate** (testi e dati).

Obiettivi del corso

- Formare **figure professionali** capaci di operare in ambienti digitali per la gestione e la pubblicazione di informazioni strutturate e non strutturate (testi e dati).

- Il corso intende mettere i partecipanti in condizione di operare a livello professionale nel mondo dell'editoria
- Forniremo quindi molte informazioni di rilevanza pratica (e un po' di teoria)
- Parte del corso sarà dedicata all'acquisizione diretta di esperienze

Obiettivi del corso

- Formare figure professionali capaci di operare in **ambienti digitali** per la gestione e la pubblicazione di informazioni strutturate e non strutturate (testi e dati).

- La gran parte dell'attività editoriale tipica si svolge oggi con l'ausilio di **mezzi elettronici/digitali**
- La gestione di **informazioni e contenuti digitali** è parte essenziale della **società della conoscenza**
- L'intero **ciclo di vita dell'informazione** è ormai digitale!

Obiettivi del corso

- Formare figure professionali capaci di operare in ambienti digitali per la **gestione** e la **pubblicazione** di informazioni strutturate e non strutturate (testi e dati).

- **Non** ci occuperemo della **generazione** dell'informazione
  - giornalismo, scrittura creativa, scrittura tecnica, ecc.
- Ci occuperemo invece della sua **gestione**:
  - trasformazione
  - presentazione
  - diffusione
  - ...

Obiettivi del corso

- Formare figure professionali capaci di operare in ambienti digitali per la gestione e la pubblicazione di **informazioni strutturate** e **non strutturate** (testi e dati).

- **Non** ci occuperemo di **contenuti** in generale
  - niente film, musica, video musicali, partite di calcio, ...
- Tratteremo invece informazione:
  - **strutturata** (basi di dati, archivi, ecc.)
  - **non strutturata** (testo libero, immagini)
- **Rappresentabile graficamente**

Obiettivi del corso

- Formare figure professionali capaci di operare in ambienti digitali per la gestione e la pubblicazione di informazioni strutturate e non strutturate (testi e dati).

- Il corso avrà un approccio **tecnologico** ai problemi citati
- Studieremo standard, algoritmi, codifiche, ...
- Altri corsi si occuperanno del **contenuto** e della **forma**
  - in particolare, questo corso si coordina con **Progettazione grafica e web design**

Programma (di massima)

- **Informazione: natura, codifica, rappresentazione**
  - codifica dei testi
    - codifiche di caratteri, codifiche di documenti
  - codifica delle immagini
    - codifiche bitmap, formati, algoritmi di compressione
    - codifiche vettoriali, formati, trasformazioni
    - codifica dei colori, modelli colore, usi tipici

Programma (di massima)

- **Tecnologie di acquisizione, memorizzazione e presentazione**
  - tecnologie di scansione, lettura ottica, importazione di documenti
  - tecnologie di memorizzazione
  - tecnologie di stampa
  - tecnologie di presentazione video
  - linguaggio Postscript e formato PDF
  - formati Office proprietari (.doc) e liberi (Open Document Format)

Programma (di massima)

- **Tipografia digitale**
  - storia, design e tecnologie per i font
  - la formattazione del testo
    - algoritmi di formattazione di paragrafo, sillabazione, formattazione di pagina
  - cenni di design e grafica
  - ambienti di word processing e desktop publishing

Programma (di massima)

- **Esercitazioni**
  - programmazione di algoritmi
    - transcodifica
    - formattazione
    - estrazione di informazione
  - disegno di font
    - per il corpo testo (leggibilità)
    - decorativi (titolazione, “dingbats”)
    - effetti speciali

Programma (di massima)

- **Esercitazioni**
  - disegno vettoriale
    - creazione di un logo
  - disegno bitmap / fotoritocco
    - “ripulitura”, fotomontaggio
  - creazione di un documento complesso
    - manuale tecnico
    - testo letterario
    - manifesto o annuncio pubblicitario
- (valutazione delle *capacità tecniche* dimostrate)

Momento buono per ulteriori domande e chiarimenti...

La codifica dei testi

- Un **testo** è una **sequenza** di **simboli**
  - Nota: un **testo** è una cosa diversa da un **documento**
- I **simboli** possono essere:
  - **caratteri** provenienti da un **alfabeto** predeterminato; hanno una rappresentazione grafica
  - **codici di controllo**, anch'essi tratti da un insieme predeterminato, solitamente non hanno rappresentazione grafica, ma alterano la rappresentazione dei caratteri

Esempio di testo

- Perché, finora cosa abbiamo proiettato??

< 'P', 'e', 'r', 'c', 'h', 'é', ',', ' ', 'f', 'i', 'n', … , '?' >

- Nota: stiamo *scrivendo* un testo che *descrive* un testo
  - ci serve un **meta-livello** descrittivo per distinguere i caratteri **di cui** parliamo da quelli **tramite i quali** parliamo
  - qui abbiamo usato il colore

L'ambigua nozione di carattere

- Spesso si tende a identificare un **carattere** ...
  - ... con una lettera dell'alfabeto
  - ... con il segno astratto che rappresenta la lettera
  - ... con la forma concreta che tale segno assume in un certo font
  - ... con l'informazione che tale segno porta
  - ... altre opzioni?

Facciamo chiarezza!

- Un **glifo,** dal greco γλύφω (gl″phō), "incidere", è un qualunque **segno** grafico (anticamente, inciso o dipinto)
- Un **grafema** è un segno elementare, non ulteriormente divisibile, del **linguaggio scritto**
- Il grafema coincide tipicamente con una **lettera** dell'alfabeto (in senso generale) del linguaggio in questione

Facciamo chiarezza!

- Un **fonema** è invece l'unità elementare del **linguaggio parlato** (suono)
- La corrispondenza fra fonemi, grafemi, lettere e glifi può essere complessa:
  - “gn” rappresenta un solo fonema, due grafemi (digrafo), due lettere
  - “fi” consta di due grafemi, ma spesso è rappresentato da un solo glifo “**ﬁ**” (legatura) che rappresenta due lettere e due fonemi (analoga legatura si ha per “ffi”)
  - “c/o” consta di due lettere e un simbolo, dunque tre grafemi, un solo glifo “℅”, una lunga serie di fonemi
  - “--” è spesso usato per indicare il trattino lungo “—“; qui due glifi codificano un solo grafema che non corrisponde a nessun fonema

Facciamo chiarezza!

- Un glifo è dunque la **rappresentazione astratta** di un grafema (es., una lettera, un ideogramma cinese), di una parte di un grafema (es., un accento) o di più grafemi (es., una legatura)
- Il grafema è una unità **di testo**
- Il glifo è una unità **grafica**

Facciamo chiarezza!

- Ad ogni glifo (unità grafica astratta) possono corrispondere molteplici disegni (unità grafica concreta)
- Esempio: glifo “A”
  - A, **A**,  , *A*,  ,  , ,  , , , , **A**, ...
- Raccolte di glifi disegnati con lo stesso stile grafico vengono dette **font** (**fonti** o **tipi**)... ma di queste ci occuperemo più avanti

Una nozione più precisa di carattere

- I **caratteri** (simboli) di cui ci occuperemo trattando della codifica dei testi sono dunque i **grafemi**
- Infatti:
  - abbiamo caratteri che non sono lettere: %
  - abbiamo caratteri che non rappresentano fonemi: !
  - abbiamo più caratteri per la stessa lettera: a, A
  - però: grafemi multi-glifo come “--” contano come due caratteri distinti

Codifica dei testi

- Com'è noto, i calcolatori elettronici sono in grado di trattare soltanto i simboli 0 e 1 (*on* e *off*, *vero* e *falso*,...)
- Sequenze di 0 e 1 costituiscono *numeri binari*; con essi si possono esprimere numeri di dimensione arbitraria
- Una codifica assegna un **codice numerico** a ogni unità testuale
  - tipicamente, **un numero per carattere**

Codifica dei testi

- I codici storicamente più importanti sono:
  - codice **ASCII** (oggi di uso universale)
  - codice **EBCDIC** (oggi quasi scomparso)
- Codici moderni basati sull'ASCII
  - **ISO-8859-xxx**
  - Windows “Code Pages” (**CP-xxx**)
- Codici “del futuro”
  - **UNICODE** (e varianti di codifica)

Il codice ASCII

- **ASCII** = **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange
- Una codifica standard creata nel 1967 (ultima revisione del 1986)
- Basato sull'alfabeto inglese:
  - niente lettere accentate
  - niente caratteri tipici di altre lingue (ß, Ç, ñ)
  - alcuni simboli tipici dell'inglese commerciale ($, %, @, &)

Il codice ASCII

- L'ASCII codifica:
  - **33** codici di controllo (oggi molti sono in disuso)
  - **95** caratteri
- In totale si hanno quindi 128 codici, che possono essere memorizzati in **7 bit** (cifre binarie) di informazione
- Visto che quasi tutti i computer lavorano con unità di 8 bit, un bit veniva lasciato libero per altri usi

Codici di controllo ASCII

0000 1010	12	10	0A	LF	^J	\n	Line feed	Nuova riga
0000 1011	13	11	0B	VT	^K		Vertical Tab	Tabulazione verticale
0000 1100	14	12	0C	FF	^L	\f	Form feed	Nuova pagina
0000 1101	15	13	0D	CR	^M	\r	Carriage return	Ritorno carrello
0000 1110	16	14	0E	SO	^N		Shift Out	
0000 1111	17	15	0F	SI	^O		Shift In	
0001 0000	20	16	10	DLE	^P		Data Link Escape	
0001 0001	21	17	11	DC1	^Q		Device Control 1 (oft. XON)	
0001 0010	22	18	12	DC2	^R		Device Control 2	
0001 0011	23	19	13	DC3	^S		Device Control 3 (oft. XOFF)	
0001 0100	24	20	14	DC4	^T		Device Control 4	
0001 0101	25	21	15	NAK	^U		Negative Acknowledgement	
0001 0110	26	22	16	SYN	^V		Synchronous Idle	
0001 0111	27	23	17	ETB	^W		End of Trans. Block	
0001 1000	30	24	18	CAN	^X		Cancel	
0001 1001	31	25	19	EM	^Y		End of Medium	
0001 1010	32	26	1A	SUB	^Z		Substitute	
0001 1011	33	27	1B	ESC	^[	\e	Escape	Codice di escape
0001 1100	34	28	1C	FS	^\		File Separator	
0001 1101	35	29	1D	GS	^]		Group Separator	
0001 1110	36	30	1E	RS	^^		Record Separator	
0001 1111	37	31	1F	US	^_		Unit Separator	
0111 1111	177	127	7F	DEL	^?		Delete	Cancella (carattere successivo)

Caratteri ASCII

Simboli, numeri, punteggiatura				Lettere maiuscole				Lettere minuscole			
Binario	Decimale	Hex	Glifo	Binario	Decimale	Hex	Glifo	Binario	Decimale	Hex	Glifo
0010 0000	32	20	*(spazio)*	0100 0000	64	40	@	0110 0000	96	60	`
0010 0001	33	21	!	0100 0001	65	41	A	0110 0001	97	61	a
0010 0010	34	22	"	0100 0010	66	42	B	0110 0010	98	62	b
0010 0011	35	23	#	0100 0011	67	43	C	0110 0011	99	63	c
0010 0100	36	24	$	0100 0100	68	44	D	0110 0100	100	64	d
0010 0101	37	25	%	0100 0101	69	45	E	0110 0101	101	65	e
0010 0110	38	26	&	0100 0110	70	46	F	0110 0110	102	66	f
0010 0111	39	27	'	0100 0111	71	47	G	0110 0111	103	67	g
0010 1000	40	28	(	0100 1000	72	48	H	0110 1000	104	68	h
0010 1001	41	29	)	0100 1001	73	49	I	0110 1001	105	69	i
0010 1010	42	2A	*	0100 1010	74	4A	J	0110 1010	106	6A	j
0010 1011	43	2B	+	0100 1011	75	4B	K	0110 1011	107	6B	k
0010 1100	44	2C	,	0100 1100	76	4C	L	0110 1100	108	6C	l
0010 1101	45	2D	-	0100 1101	77	4D	M	0110 1101	109	6D	m
0010 1110	46	2E	.	0100 1110	78	4E	N	0110 1110	110	6E	n
0010 1111	47	2F	/	0100 1111	79	4F	O	0110 1111	111	6F	o
0011 0000	48	30	0	0101 0000	80	50	P	0111 0000	112	70	p
0011 0001	49	31	1	0101 0001	81	51	Q	0111 0001	113	71	q
0011 0010	50	32	2	0101 0010	82	52	R	0111 0010	114	72	r
0011 0011	51	33	3	0101 0011	83	53	S	0111 0011	115	73	s
0011 0100	52	34	4	0101 0100	84	54	T	0111 0100	116	74	t
0011 0101	53	35	5	0101 0101	85	55	U	0111 0101	117	75	u
0011 0110	54	36	6	0101 0110	86	56	V	0111 0110	118	76	v
0011 0111	55	37	7	0101 0111	87	57	W	0111 0111	119	77	w
0011 1000	56	38	8	0101 1000	88	58	X	0111 1000	120	78	x
0011 1001	57	39	9	0101 1001	89	59	Y	0111 1001	121	79	y
0011 1010	58	3A	:	0101 1010	90	5A	Z	0111 1010	122	7A	z
0011 1011	59	3B	;	0101 1011	91	5B	[	0111 1011	123	7B	{
0011 1100	60	3C	<	0101 1100	92	5C	\	0111 1100	124	7C	|
0011 1101	61	3D	=	0101 1101	93	5D	]	0111 1101	125	7D	}
0011 1110	62	3E	>	0101 1110	94	5E	^	0111 1110	126	7E	~
0011 1111	63	3F	?	0101 1111	95	5F	_				

Esempio di codifica ASCII

- Vediamo come viene codificata la frase “Ciao, mamma!”:

Carattere	C	i	a	o	,		m	a	m	m	a	!
Codice	67	105	97	111	44	32	109	97	109	109	97	33

- È importante ricordare che il calcolatore “vede” solo i codici numerici
- Serve un altro sistema per sapere quale codice si sta usando
  - tipicamente, è dato implicitamente
  - a volte, si specifica l'*encoding* in testa al documento

Codifica di testi in ASCII

- Un testo ASCII è codificato da una serie di caratteri, con interposti codici di controllo
- Spesso, gli unici codici di controllo usati sono quelli che indicano gli a-capo
  - ad ogni fine riga, se il testo è **preformattato** (*a-capo fisici*)
  - solo a fine paragrafo, se il testo **non è formattato** (*a-capo logici*)
    - in questo caso, il programma che visualizza il testo dovrà riformattarlo al volo

Codifica di testo in ASCII

- **Attenzione** a una trappola comune:
  - UNIX (Linux) e molti altri sistemi usano il codice **LF** (13) per indicare a-capo
  - Macintosh usa il codice **CR** (10)
  - Windows e MS-DOS usano entrambi i codici, ovvero la sequenza **CR+LF** (10+13)
- Se il testo è stato generato su un S.O. diverso e programmi usati non fanno la conversione automaticamente, può capitare di avere a che fare con linee **estremamente** lunghe!

Il codice EBCDIC

- Prodotto da IBM (tentando di competere con ASCII) nel 1963-64
- Usato su tutti i *mainframe* IBM dal System/360 in poi (tranne che da Linux su zSeries)
- Codice a 8 bit, contiene più caratteri rispetto all'ASCII, ma...
  - poco standardizzato
  - codifica bislacca (alfabeto disgiunto!)
- Interesse solo storico

Il codice EBCDIC

Hex	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
4-			â	ä	à	á	ã	å	ç	ñ	[	.	<	(	+	!
5-	&	é	ê	ë	è	í	î	ï	ì	ß	]	$	*	)	;	^
6-	-	/	Â	Ä	À	Á	Ã	Å	Ç	Ñ	¦	,	%	_	>	?
7-	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	@	'	=	"
8-	Ø	**a**	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	«		»	ð	″	þ	±
9-	°	**j**	**k**	**l**	**m**	**n**	**o**	**p**	**q**	**r**	ª	º	æ	¸	Æ	¤
A-	µ	~	**s**	**t**	**u**	**v**	**w**	**x**	**y**	**z**	¡	¿	Ð	Ý	Þ	®
B-	¢	£	¥	·	©	§	¶	¼	½	¾	¬	|	¯	¨	´	×
C-	**{**	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	-	ô	ö	ò	ó	õ
D-	**}**	**J**	**K**	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	¹	û	ü	ù	ú	ÿ
E-	\	÷	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**	²	Ô	Ö	Ò	Ó	Õ
F-	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	³	Û	Ü	Ù	Ú	

CCSID500, una delle varianti di EBCDIC

Svantaggi di ASCII e EBCDIC

- ASCII, limitato a 95 caratteri, non era sufficiente per linguaggi diversi dall'Inglese (e anche per quello...)
  - Es: perche' invece di perché
- EBCDIC, mai standardizzato davvero, era inutilizzabile come formato di scambio fra macchine diverse
  - persino il codice di “a capo” poteva essere diverso su macchine diverse!