



UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI ...

CORSO DI LAUREA ...
IN INGEGNERIA ...

Titolo tesi

Titolo inglese

Relatore:

CH.MO PROF. NOME COGNOME

Laureando:

NOME COGNOME

1234567

Anno Accademico 20XX/20XX

Inserisci qui la tua citazione o dedica.

Ringraziamenti

Inserisci gli eventuali ringraziamenti personali.

Sommario

Inserire abstract.

Introduzione

Scrivi qui la tua introduzione.

Indice

Elenco delle figure	xi
Elenco delle tabelle	xiii
Elenco dei codici	xv
1 Primo Capitolo	1
1.1 Sezione 1	1
1.2 Sezione 2	1
1.2.1 Sottosezione	1
2 Immagini e Tabelle	3
2.1 Immagine singola	3
2.2 Immagine multipla	3
2.3 Tabelle	4
3 Formule	7
4 Pseudocodice e codice	9
4.1 Pseudocodice	9
4.2 Codice	10
Conclusioni	13
Bibliografia	15
Sitografia	17

Elenco delle figure

2.1	Didascalia	3
2.2	Esempio di figura composta da 4 figure.	4

Elenco delle tabelle

2.1	Tabella 1	4
2.2	Tabella 2	5

Elenco dei codici

4.1	Didascalia.	10
4.2	prova.cpp	10

Capitolo 1

Primo Capitolo

La struttura utilizzata in questo template non è obbligatoria, però ritengo che sia molto comoda per evitare di scrivere file troppo lunghi e di avere un controllo migliore sulla struttura. Questa prevede di scrivere l'introduzione al capitolo in un file salvato nella cartella principale e di sviluppare le sezioni all'interno di una cartella. Esempio di richiamo ad un riferimento [1].

1.1 Sezione 1

Ad ogni sezione, in questo template [2], corrisponde un file all'interno della cartella relativa al capitolo [3].

1.2 Sezione 2

Una sezione può contenere una sottosezione. In questo caso, è stato deciso di non creare altri file... [4]

1.2.1 Sottosezione

...ma di scrivere il testo all'interno del file della sezione corrente [5].

Capitolo 2

Immagini e Tabelle

Metodi più comuni per inserire immagini e esempi di tabelle. [6]

2.1 Immagine singola

Per fare riferimento ad un immagine [7], come ad ogni altro elemento a cui viene attribuita una *label* è disponibile il comando *ref.* Riferimento a Figura 2.1.

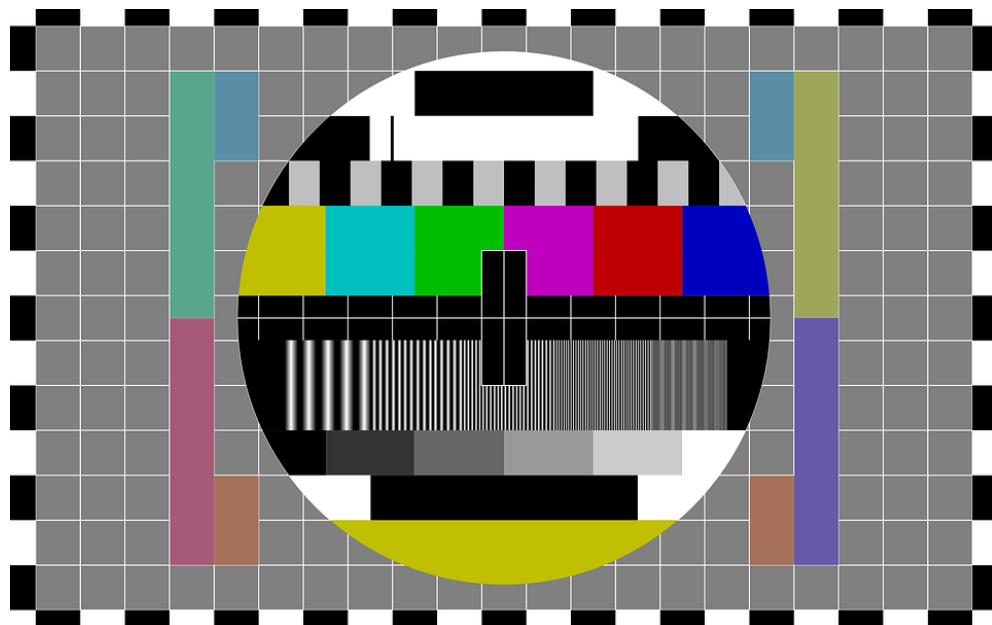


Figura 2.1: Didascalia

2.2 Immagine multipla

Inserire più “sottofigure” in una figura.

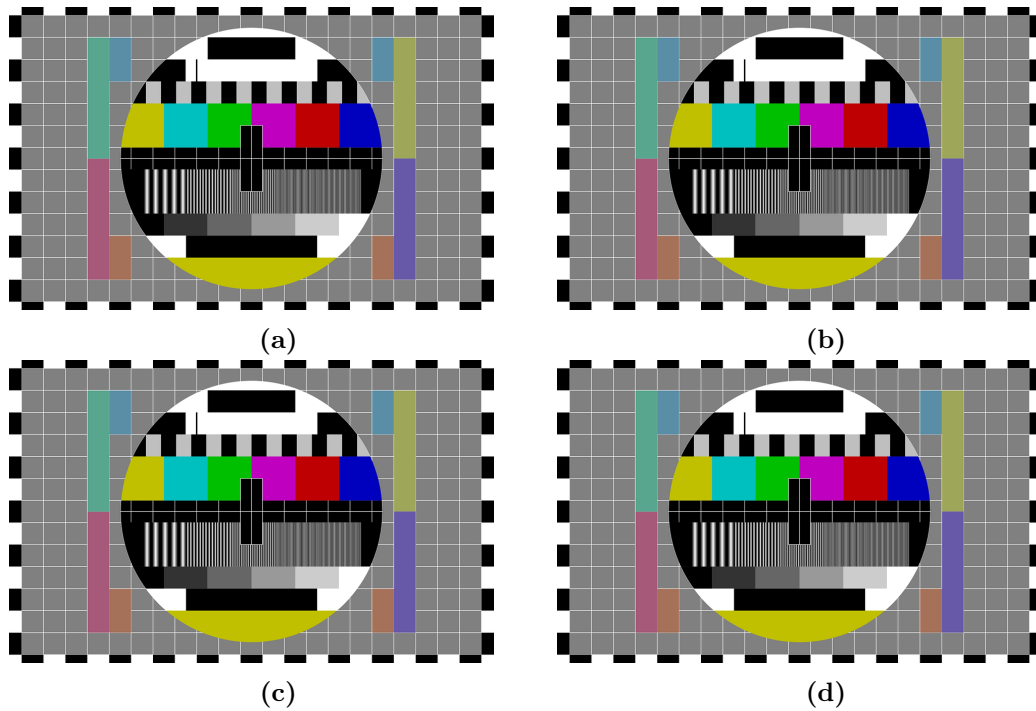


Figura 2.2: Esempio di figura composta da 4 figure.

2.3 Tabelle

Nella seguente tabella vengono mostrati alcuni esempi di separazione delle righe. La separazione delle colonne avviene all'interno delle parentesi graffe dopo il comando *tabular*.

cella1	cella2	cella3
cella4	cella5	cella6
cella7	cella8	cella9
cella10	cella11	cella12

Tabella 2.1: Tabella 1

È possibile, inoltre, fissare la dimensione delle colonne [8].

cella1	cella2	cella3
cella4	cella5	cella6 cella6 cella6 cella6
cella7 cella10	cella8 cella11	cella9 cella12

Tabella 2.2: Tabella 2

Capitolo 3

Formule

Per inserire delle formule matematiche è possibile utilizzare due metodi [9]:

- in linea: inserendo la formula tra due caratteri \$.
- utilizzando l'ambiente *equation*.

Esempio di formula in linea $x(t) = x_0 + v_0t + \frac{1}{2}at^2$.

Esempio di utilizzo dell'ambiente *equation*:

$$x(t) = x_0 + v_0t + \frac{1}{2}at^2 \tag{3.1}$$

Possiamo usare le *label* anche per le equazioni [10]. Legge oraria nell'Equazione 3.1. Infine, un esempio di formula su più righe:

$$\begin{aligned} x(t) &= x_0 + v_0t + \frac{1}{2}at^2 \\ &= x_0 + v_0t + \frac{1}{2}\frac{F}{m}t^2 \end{aligned} \tag{3.2}$$

Capitolo 4

Pseudocodice e codice

In questo template per l’inserimento di pseudocodice è stato utilizzato il pacchetto *algpseudocode*. Per quanto riguarda l’inserimento del codice è possibile utilizzare il comando *verb* per inserire in linea oppure *lstlisting* per inserire blocchi di codice [11].

4.1 Pseudocodice

Algorithm 1 Nome algoritmo

Require: Input dell’algoritmo

Ensure: Output dell’algoritmo

```
1: variabile  $\leftarrow$  assegnazione valore
2: valore_ritornato  $\leftarrow$  FUNZIONEPROVA(param1, param2)
3: for element in list do
4:   res  $\leftarrow$  DOSOMETHING(element)
5: end for
6: if condizione1 then
7:   do something
8: else if condizione2 then
9:   do something else
10: else
11:   print "Hello World"
12: end if
```

4.2 Codice

È possibile inserire codice in linea: `print("Hello World")` [12].

Inoltre è possibile usare l'ambiente *lstlisting* configurando il layout del blocco di codice nel file *layout.tex*. Il codice può essere importato da un file esterno che metteremo nella cartella *code* [13].

```

1  # Number of trees in random forest
2  n_estimators = [int(x) for x in np.linspace(start = 100, stop = 1000, num =
    10)]
3  # Number of features to consider at every split
4  max_features = ['log2', 'sqrt']
5  # Maximum number of levels in tree
6  max_depth = [int(x) for x in np.linspace(10, 150, num = 15)]
7  max_depth.append(None)
8  # Minimum number of samples required to split a node
9  min_samples_split = [2, 5, 10]
10 # Minimum number of samples required at each leaf node
11 min_samples_leaf = [1, 2, 4, 6, 8]
12 # Method of selecting samples for training each tree
13 bootstrap = [True, False]
14 random_grid = {'n_estimators': n_estimators,
15                'max_features': max_features,
16                'max_depth': max_depth,
17                'min_samples_split': min_samples_split,
18                'min_samples_leaf': min_samples_leaf,
19                'bootstrap': bootstrap
20                }
21
22 rfc = RandomForestClassifier()
23 rfc_random = RandomizedSearchCV(estimator=rfc, param_distributions=random_grid
    , n_iter=1000, cv=5, verbose=2, random_state=3, n_jobs=4)
24
25 rfc_random.fit(df_downsampled, labels_downsampled)

```

Codice 4.1: Didascalìa.

```

1  // comment
2  bool Data::DoStuff(int enum_type){
3      Param* par=NULL;
4      for(object=obj.begin();obj<obj.end();++obj){
5          if(par->GetEnum()==enum_type){ return true; }
6      }
7      _error_("string here!" << strx(enum_type));

```

```
8     return false;  
9 }
```

Codice 4.2: prova.cpp

Conclusioni

Scrivi qui le tue conclusioni.

Bibliografia

Capitolo 1

- [1] P. Vas. *Sensorless Vector and Direct Torque Control*. pp. 407–410. Oxford University Press, 1998 (cit. a p. 1).
- [3] S.Bolognani e M. Zigliotto. «Self-Commissioning Compensation of Inverter Non-Idealities for Sensorless AC Drives Applications». In: *Proceedings of IEE International Conference on Power Electronics, Machines and Drives (PEMD 2002)*. Bath, UK, giu. 2002, pp. 30–37 (cit. a p. 1).
- [4] Lapo Filippo Mori. *Scrivere poesie con LaTeX*. <https://www.guitex.org/home/>. 2007 (cit. a p. 1).
- [5] A. R. Munoz e T. A. Lipo. «On-Line Dead-Time Compensation Technique for Open-Loop PWM-VSI Drives». In: *IEEE Trans. Power Electron.* 14.4 (lug. 1999), pp. 683–689 (cit. a p. 1).

Capitolo 2

- [6] Umberto Eco. *Come si fa una tesi di laurea*. Milano: Bompiani, 1977 (cit. a p. 3).
- [7] Paul Adrien Maurice Dirac. *The Principles of Quantum Mechanics*. International series of monographs on physics. Clarendon Press, 1981. ISBN: 9780198520115 (cit. a p. 3).

Capitolo 3

- [10] *LaTeX su Wikipedia*. <https://it.wikipedia.org/wiki/LaTeX>. 2017 (cit. a p. 7).

Capitolo 4

- [11] Albert Einstein. «Zur Elektrodynamik bewegter Körper. (German) [On the electrodynamics of moving bodies]». In: *Annalen der Physik* 322.10 (1905), pp. 891–921. DOI: <http://dx.doi.org/10.1002/andp.19053221004> (cit. a p. 9).
- [13] Lapo Filippo Mori. «Scrivere la tesi di laurea con LaTeX». In: *Ars Volume.3* (2007) (cit. a p. 10).

Sitografia

Capitolo 1

- [2] *Usability first Scientific visualization definition*. URL: <http://www.usabilityfirst.com/glossary/scientific-visualization/> (cit. a p. 1).

Capitolo 2

- [8] *prova link2*. URL: <http://www.google.com> (cit. a p. 4).

Capitolo 3

- [9] *prova link1*. URL: <http://www.blank.org> (cit. a p. 7).

Capitolo 4

- [12] Donald Knuth. *Knuth: Computers and Typesetting*. URL: <http://www-cs-faculty.stanford.edu/~uno/abcde.html> (cit. a p. 10).