



IN QUESTO NUMERO:

la rappresentazione interna del codice operativo LOAD mentre 0100 è il numero 8 in binario.
Se si voleva scrivere, ad esempio, la frase "Ciao Filippo" sullo schermo si scriveva qualcosa del tipo:

```
00000000 7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00 00
00000010 02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 00
00000020 c8 00 00 00 00 00 00 00 34 00 20 00 02 00 28 00
00000030 04 00 03 00 01 00 00 00 00 00 00 00 00 80 04 08
00000040 00 80 04 08 9d 00 00 00 9d 00 00 00 05 00 00 00
00000050 00 10 00 00 01 00 00 00 a0 00 00 00 a0 90 04 08
00000060 a0 90 04 08 0e 00 00 00 0e 00 00 00 06 00 00 00
00000070 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000080 ba 0e 00 00 00 b9 a0 90 04 08 bb 01 00 00 00 b8
00000090 04 00 00 00 cd 80 b8 01 00 00 00 cd 80 00 00 00
000000a0 43 69 61 6f 20 46 69 6c 69 70 70 6f 00 00 00 00
000000b0 73 68 73 74 72 74 61 62 00 2e 74 65 78 74 00 2e
000000c0 64 61 74 61 00 00 00 00 00 00 00 00 00 00 00 00
000000d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000e0 0b 00 00 00 01 00 00 00 06 00 00 00 80 80 04 08
```

TUTORIAL

Introduzione alla programmazione
parte I°



INFORMATICA

Siti web ikea vs artigianali



LUOGHI

La Base Tuono (Tn)



CUCINA

Ravioli di gamberetti
al profumo d'agrumi

INTRODUZIONE ALLA PROGRAMMAZIONE

I° PARTE

COSA SONO I PROGRAMMI

Un programma non è altro che una serie di istruzioni, scritte in uno specifico linguaggio, da fare eseguire ad un computer.

Facciamo l'esempio di dover dare delle indicazioni stradali ad una persona. Diremmo qualcosa del tipo:

Vai avanti fino al semaforo, gira a destra e prosegui per tutta la via. Quando arrivi in piazza parcheggia che sei arrivato.

Ecco! Un programma si comporta allo stesso modo.

Nel nostro esempio abbiamo dato le istruzioni in Italiano, ma nel caso il nostro interlocutore fosse stato, ad esempio, inglese non avremmo potuto dirle in italiano ma nella sua lingua o in una lingua che entrambi conosciamo.

Anche nel mondo dei computer esistono tante lingue per fare i programmi e come nel mondo reale ogni lingua ha una sua grammatica, un suo lessico ed una sua sintassi e costruzione logica delle frasi e dei "dialetti".

TIPOLOGIA DI LINGUAGGI DI PROGRAMMAZIONE

Per "comunicare" con i computer i primi programmatori usavano scrivere i comandi direttamente nella lingua dei computer 01010110. Ecco che, ad esempio, scrivere 00110100 poteva voler dire "LOAD 8" dove 0011 è la rappresentazione interna del codice operativo LOAD mentre 0100 è il numero 8 in binario.

Se si voleva scrivere, ad esempio, la frase "Ciao Filippo" sullo schermo si scriveva qualcosa del tipo:

```
00000000 7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00
00000010 02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 00
00000020 c8 00 00 00 00 00 00 00 34 00 20 00 02 00 28 00
00000030 04 00 03 00 01 00 00 00 00 00 00 00 08 04 08
00000040 00 80 04 08 9d 00 00 00 9d 00 00 00 05 00 00 00
00000050 00 10 00 00 01 00 00 00 a0 00 00 00 a0 90 04 08
00000060 a0 90 04 08 0e 00 00 00 0e 00 00 00 06 00 00 00
00000070 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000080 ba 0e 00 00 00 b9 a0 90 04 08 bb 01 00 00 00 b8
00000090 04 00 00 00 cd 80 b8 01 00 00 00 cd 80 00 00 00
000000a0 43 69 61 6f 20 46 69 6c 69 70 70 6f 00 00 00 00
000000b0 73 68 73 74 72 74 61 62 00 2e 74 65 78 74 00 2e
000000c0 64 61 74 61 00 00 00 00 00 00 00 00 00 00 00 00
000000d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
000000f0 0b 00 00 00 01 00 00 00 06 00 00 00 80 80 04 08
00000100 80 00 00 00 1d 00 00 00 00 00 00 00 00 00 00 00
00000110 10 00 00 00 00 00 00 00 11 00 00 00 01 00 00 00
00000120 03 00 00 00 a0 90 04 08 a0 00 00 00 0e 00 00 00
00000130 00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00
00000140 01 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00
00000150 ae 00 00 00 17 00 00 00 00 00 00 00 00 00 00 00
00000160 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Dove la linea scritta in rosso corrisponde a "Ciao Filippo".

Questo tipo di programmazione era molto difficile ed era molto vincolato al tipo di macchina usato.

Il passaggio successivo fu quello di "semplificare" questo metodo di programmazione con l'introduzione dell'assembly, un linguaggio molto simile al linguaggio macchina ma "più semplice" anche se sempre legato al tipo di hardware.

Per scrivere un programma in assembly si scrive un file di testo che poi viene dato in pasto ad un altro programma (chiamato assembler) che restituisce un eseguibile (ovvero un programma che il computer può eseguire); a seconda dell'assembler utilizzato si doveva usare uno specifico "dialetto".

Ciao Filippo in assembly per il compilatore NASM

```
org 0x100
mov dx, msg
mov ah, 9
int 0x21
mov ah, 0x4c
int 0x21
msg db 'Ciao Filippo', 0x0d, 0x0a, '$'
```

Ciao Filippo in assembly per il compilatore INTEL

```
Model small
Stack 100h
.data
hw db "ciao Filippo", 13, 10, '$'
.code
.startup
mov ax, @data
mov ds, ax
mov dx, offset hw
mov ah, 09h
int 21h
mov ax, 4c00h
int 21h
End
```

Come si vede entrambi i metodi sono molto simili, fanno riferimento agli stessi indirizzi di memoria e funzioni ma sono scritti in un "dialetto" differente l'uno dall'altro: per entrambi `mov dx` sposta il messaggio nell'indirizzo di memoria `dx`, `mov ah,9` richiama la funzione per stampare, `int 0x21` richiamo il servizio DOS, `db` contiene il messaggio e `'$'` serve a indicare che il messaggio è terminato; `Stack 100h` e `org 0x100` indicano, che il programma deve essere caricato all'indirizzo `0x0100` in quanto compilando il programma diventerà il programma `ciao.com` (e non `exe!`). Le differenze sono nella sintassi. Da un punto di vista "grammaticale" invece vediamo che per scrivere un testo in NASM lo sacchiudo tramite gli apici (`' '`) mentre in INTEL tramite le virgolette (`" "`).

Con la crescita dell'uso dei computer ci si accorse ben presto che i linguaggi assembly facevano risparmiare tempo rispetto al linguaggio macchina, ma che, se andavano bene per i comandi base di un calcolatore, diventava complesso gestire un programma troppo evoluto.

Tra gli anni '50 e '60 del XX° secolo si iniziarono a sviluppare linguaggi più complessi, che risultavano più facili da apprendere e da usare e che facevano risparmiare tempo ai programmatori. Si preferì cioè dare priorità alla realizzazione di un programma piuttosto che alla sua esecuzione.

Nascono così quelli che vengono chiamati "linguaggi di alto livello": più un linguaggio si avvicina al linguaggio macchina più viene chiamato di basso livello, mentre più si allontana più è di alto livello.

I linguaggi di alto livello necessitano un compilatore o un interprete che sia in grado di "tradurre" le istruzioni del linguaggio di alto livello in istruzioni macchina di basso livello eseguibili dal computer.

Un compilatore è simile ad un assemblatore ma molto più complesso perché ogni singola istruzione di un linguaggio di alto livello corrisponde a più istruzioni in linguaggio macchina e quindi deve "tradurre" in maniera più complicata. Il risultato sarà un programma eseguibile (solitamente `.exe`).

Un interprete, invece, è un programma che è in grado di tradurre un file (solitamente di testo) contenente un programma scritto in un linguaggio di programmazione, e di farlo eseguire dal computer.

Come conseguenza avremmo che un programma interpretato sarà molto più lento nell'esecuzione di un programma compilato.

Linguaggi di alto livello sono, ad esempio, il Pascal, il C ed il Basic, molto usati negli anni '70.

Ciao Filippo in PASCAL

```
program ciaoFilippo;
begin
  writeln('Ciao Filippo');
  readln;
end.
```

Ciao Filippo in C

```
#include <stdio.h>
int main()
{
  printf("Ciao Filippo");
  return 0;
}
```

Ciao Filippo in BASIC

```
CLS
PRINT "Ciao Filippo"
```

Come si vede, scrivere un programma in questo modo semplifica, e di molto, sia i passaggi necessari sia un eventuale modifica.

Negli ultimi anni si è sviluppato un nuovo approccio ai linguaggi di programmazione, l'uso di bytecode.

Il bytecode è un linguaggio intermedio: il codice sorgente viene compilato in un formato intermedio (chiamato appunto bytecode), il quale a sua volta viene interpretato da una Virtual Machine, che ha il compito di interpretare "al volo" le istruzioni bytecode in istruzioni per il processore. Questa metodologia permette di creare programmi che hanno un grande livello di portabilità e al contempo velocità. Basta avere la macchina virtuale installata sul proprio computer (non importa se il sistema sia Windows, Mac, Linux o altro) e il programma bytecode verrà eseguito.

La macchina virtuale più famosa è sicuramente la Java Virtual Machine che "traduce" il bytecode (solitamente un file .class) in linguaggio macchina tramite una procedura chiamata just in time.

Anche la microsoft (tramite il frameworks .Net) ha creato il suo strumento di sviluppo, anche più complesso ed articolato di quello di Sun, dato che .Net può interagire con diversi linguaggi, il più famoso ed utilizzato dei quali è C# (# si legge sharp).

Ciao Filippo in JAVA

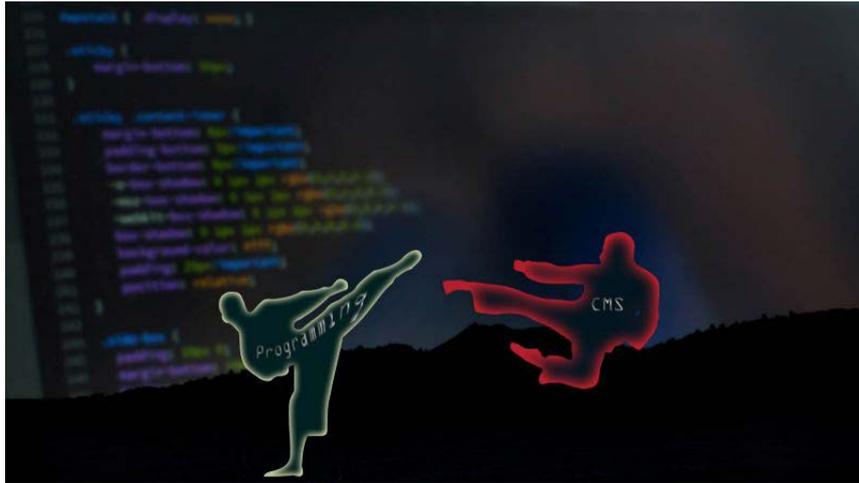
```
public class ciao_filippo {  
  
    public static void main(String[] args) {  
        System.out.println("Ciao Filippo");  
    }  
}
```

Ciao Filippo in C#

```
using System;  
class ciao  
{  
    static void Main()  
    {  
        Console.WriteLine("Ciao Filippo");  
    }  
}
```

GUIDA COMPLETA SU: <https://filoweb.it/appunti/Introduzione-alla-programmazione.pdf>

SITI WEB IKEA VS ARTIGIANALI



Quando si decide di fare un sito web, per prima cosa bisogna decidere il tipo di “strumento” da utilizzare. Principalmente ci si orienta tra l'affidarsi ad un CMS e l'affidarsi a un Frameworks¹ (tralasciamo i classici siti web statici fatti solamente in HTML e CSS in quanto obsoleti). Per capire di cosa stiamo parlando iniziamo con il dire che i CMS (Sistemi di gestione dei contenuti) riducono, per l'utente che lo usa, la necessità di avere conoscenze specifiche di programmazione, mentre un frameworks fornisce una serie di funzionalità che è possibile utilizzare per creare un sito personalizzato.

Cosa sono i CMS

I CMS, come abbiamo detto sopra, sono un sistema che permette di gestire i contenuti senza la necessità di una conoscenza specifica. Tra i Content Management Systems più famosi spiccano Wordpress, Joomla!, Drupal, Magneto, Wix. Le caratteristiche di ogni CMS sono diverse ma il loro nucleo e metodo di utilizzo è suppergiù il medesimo:

l'utente installa sul suo server il CMS tramite semplici passaggi (solitamente 1 o 2)

sceglie un template² tra quelli disponibili

Inserisce i contenuti

Attiva eventuali Plug-in³

Da questi passaggi si evince che un CMS non necessita di nessuna conoscenza di programmazione in quanto si basa su moduli preinstallati.

Cos'è un Frameworks

Un framework è un'architettura logica di supporto su cui un software può essere progettato e realizzato (quindi non è un linguaggio di programmazione!). La sua funzione è quella di creare una infrastruttura generale, lasciando al programmatore il contenuto vero e proprio dell'applicazione. Lo scopo di un framework è infatti quello di risparmiare allo sviluppatore la riscrittura di codice già scritto in precedenza per compiti simili.

Un Framework permette di interagire con un linguaggio di programmazione (e, come nel caso di .Net, di compilarne il codice).

I frameworks più famosi per la creazione di applicazioni web sono:

- .Net, che può interagire con diversi linguaggi che vanno dal C# al Vb.Net

- Laravel e Symfony, che interagisce con PHP

- Angular e Node.js, ottimi per Javascript

- Django, creato per Python

- Ruby on Rails, ottimo per usare Ruby

Si capisce subito che utilizzare un Frameworks richiede conoscenze specifiche e necessita di più tempo per la realizzazione di un progetto.

Personalizzazione

A chi utilizza principalmente i CMS per creare siti web piace molto il motto “perché reinventare la ruota se esiste già?” per riferirsi al fatto che per i CMS esistono centinaia di migliaia di templates disponibili (gratis o a pagamento) tra i quali scegliere e quindi non ha senso sforzarsi nel creare del nuovo codice e dei nuovi layout per un progetto web. In realtà la maggior parte di chi utilizza i CMS non ha molte conoscenze di programmazione e tende a nascondersi dietro questo modo di pensare.

Tramite un CMS, infatti si hanno a disposizione, come già accennato sopra, tutta una serie di strumenti, tools e plug-in tra i quali scegliere e facilmente installabili e configurabili ma...

Potrebbero insorgere complicazioni quando è necessario che il sito web si adatti alle proprie esigenze specifiche. Con un framework, invece, si deve creare un progetto da zero, il che porta a realizzare caratteristiche uniche e distintive: i framework sono altamente personalizzabili mentre un CMS ha in genere dei limiti. Con un CMS, ad esempio, non è possibile modificare la funzionalità di base o non si aggiornerà correttamente, mentre un framework non ha limiti! La realizzazione di un progetto web tramite un framework anziché di un CMS comporta, quindi, dei tempi di lavorazione più lunghi (tenendo conto della progettazione, debug⁴, ecc.) ma un grado di personalizzazione maggiore.

Sicurezza

Sucuri (azienda leader nel monitoraggio e protezione dei siti web) ha rilevato che nel 2018 la maggior parte dei siti web hackerati (90%) erano fatti tramite Wordpress (che è il CMS più utilizzato), seguiti da Magento e Joomla!. Gli esperti hanno rilevato che la maggior parte degli attacchi avvenivano tramite plug-in o non aggiornamenti dei sistemi stessi.

Il motivo di questa grande quantità di attacchi è presto chiarita: i CMS ed i relativi Plug-in, sono per la maggior parte progetti Open Source⁵ e quindi chiunque ne può vedere il codice sorgente ed eventualmente le vulnerabilità (così come chiunque può correggerle!)

Un progetto che utilizza un framework, invece, è quasi sempre scritto tramite un codice personalizzato e specifico da chi lo crea e quindi (a parte le vulnerabilità delle infrastrutture) risulta meno soggetto ad attacchi hacker.

Un framework ben sviluppato è molto più sicuro di un CMS generico, anche se i sistemi di gestione dei contenuti hanno spesso plug-in scritti per rafforzare la loro sicurezza.

Flessibilità

Indipendentemente dal tipo di progetto che si vuole creare può nascere la necessità di dover integrare lo stesso con delle funzionalità non previste come ad sistemi di CRM (customer relationship management ovvero la gestione dei rapporti con i clienti) che possano interagire con altri strumenti e/o siti, come nel caso di Salesforce⁶.

Un progetto sviluppato tramite un framework sarà benissimo in grado di integrarsi e modellarsi con ogni necessità, mentre un progetto creato tramite un CMS dovrà "adattarsi" in base ai plug-in disponibili oppure affidarsi alla realizzazione dei plug-in personalizzati tramite un programmatore o un'agenzia con costi molto elevati.

Certo esiste anche la possibilità di creare progetti personalizzati che si basano sul core di un CMS ma in questo caso i costi partono, spesso, dai 5.000 euro superando, a volte, i 10.000, quindi non molto lontani (a volte anzi superiori) da quelli della realizzazione di una piattaforma web tramite l'uso di un framework.

Quando si parla di flessibilità, quindi, un framework offre un grado di libertà a 360° a fronte di costi di sviluppo più alti, mentre un CMS risulta molto più limitato con un costo minore.

Cosa scegliere?

I CMS ed i Frameworks sono strumenti completamente diversi e la scelta di utilizzare l'uno anziché l'altro dipende da vari fattori. Prima di scegliere quale metodo di sviluppo scegliere è bene considerare alcuni punti:

La quantità di tempo che si è disposti a investire su di un progetto

La quantità di soldi che si possono/vogliono investire sullo stesso

L'unicità del progetto ed il suo grado di flessibilità

Il target al quale il progetto è destinato

Il grado di sicurezza di cui si necessita

Sicuramente per il blog personale che non necessita di troppe visite o anche di un semplice negozio sotto casa l'utilizzo di un CMS può rappresentare una scelta oculata, laddove, nel caso ci si affidi ad un'azienda o a un professionista, i costi non siano troppo elevati.

Se il progetto riguarda invece un progetto web che deve interagire con i clienti, presentare prodotti e come target si designi traguardi elevati è meglio iniziare subito con un Framework (meglio se .net) onde evitare poi di dover ricominciare tutto da capo.

Ricordiamo, infine, che pochissimi degli utenti che accederanno ad un sito web sono in grado di capire con quale tecnologia sia stato sviluppato, ma prestano più attenzione alla fruibilità del sito, ai contenuti e a quanto sia conforme alle sue necessità.

¹ Per una spiegazione di cosa sia un framework si consiglia di leggere il precedente articolo "Php o .Net? Facciamo Chiarezza" - NEWSLETTER N°1

² Il template nel web è un modello usato per separare il contenuto dalla presentazione grafica, e per la produzione in massa di pagine Web.

³ Con il termine plug-in, in questo caso, ci si riferisce a un programma non autonomo che interagisce con un altro programma per ampliarne o estenderne le funzionalità originali.

⁴ Nella programmazione informatica il debug è l'attività di ricerca e correzione degli errori (bug) in un programma.

⁵ Letteralmente "sorgente aperta", indica un software il cui codice sorgente è rilasciato con una licenza che lo rende modificabile o migliorabile da parte di chiunque.

⁶ Salesforce è la più importante piattaforma per la gestione delle relazioni con i clienti (CRM)

LA BASE TUONO

FOLGARIA (TN)

Un museo originale, unico in Europa



Situata tra cime delle montagne trentine, prospiciente un laghetto artificiale, a Passo Coe (1600 s.l.m.) si trova la Base Tuono, un'ex base militare missilistica, del tempo della guerra fredda rimasta in attività dal 1966 al 1977.

Iniziato nel 2010 la costruzione del museo è un monumento al periodo in cui si pensava che un conflitto tra super potenze potesse avvenire. Nel 2013 la Base Tuono ha ufficialmente ottenuto il riconoscimento come sito museale rappresentativo del sistema d'arma Nike-Hercules dallo Stato Maggiore dell'Aeronautica; all'interno del perimetro che delimita la base si trovano esposti, infatti, gli ultimi missili terra-aria Nike-Hercules (ovviamente disarmati) esistenti in Italia.

Per chi visita la base è consigliato approfittare delle visite guidate che non solo spiegano il sistema d'ar-

ma di cui facevano parte i missili Nike-Hercules, ma racconta anche il clima geo-politico che ha condizionato la storia mondiale per più quasi mezzo secolo. La prima parte della visita inizia, infatti, con un'introduzione storica che illustra il periodo della Guerra Fredda nel contesto internazionale per proseguire con il sistema di difesa missilistico italiano, il dispiegamento delle basi missilistiche, l'operatività dei missili Nike-Hercules e le caratteristiche tecniche di questi ultimi.



RAVIOLI DI GAMBERETTI AL PROFUMO D'AGRUMI

Primo Piatto

Ingredienti:

250 gr di gamberetti	1 spicchio d'aglio
200 gr di farina 00 (per l'impasto)	1/2 bicchiere vino bianco
1 cucchiaio di farina (per il condimento)	prezzemolo q.b.
2 uova	sale q.b.
1 arancia	1 noce di burro
1/2 limone	Olio d'oliva q.b.



Procedimento:

Con l'avvicinarsi dell'estate un piatto che rinfrescherà le serate con gli amici: i ravioli di gamberetti al profumo d'agrumi

Per il ripieno:

In una padella mettere a scaldare un cucchiaio di olio d'oliva con 1 spicchio d'aglio. Quando inizia a soffriggere l'aglio aggiungere i gamberetti (sgusciati), il prezzemolo a piacere, salare e sfumare con il vino bianco.

Quando saranno cotti tritarli e mettere il tutto a riposare.

Per la pasta

Per preparare la pasta bisogna ricordare che ci vogliono circa 100gr di farina per ogni uovo, in questa ricetta la porzione è per 2 persone quindi abbiamo usato 200gr di farina e due uova.

Mettere la farina "00" sul tavolo o su di una spianatoia formando una fontana, con le dita si crea una cavità simile a un cratere. Si passa poi a spolverare il tutto con un po' di sale e ci si sguscia al centro della farina le uova.

Si inizia poi ad incorporare un po' di farina presa dai bordi del cratere e si prosegue a lavorare gli ingredienti con le dita fino a quando non si sarà raccolta tutta la farina.

Dopo aver impastato almeno 10 minuti in modo energico si forma una palla, la si avvolge nella pellicola e la si lascia riposare l'impasto per almeno 10-15 minuti.

Trascorso il tempo si prende l'impasto lo si appiattisce con le mani per conferirgli una forma a disco e lo si stende sul tavolo o sulla spianatoia (sempre ben infarinati).

Con l'aiuto di un mattarello partendo sempre dal centro e facendolo scorrere in tutte le direzioni in modo da ottenere una sfoglia dallo spessore uniforme la si spiana. Quando la sfoglia comincia ad allargarsi la si gira, avvolgendola sul mattarello e la si ruota di 90°.

Continuare a stendere la sfoglia fino a raggiungere lo spessore desiderato.

Una volta ottenuta una bella sfoglia con l'aiuto di un bicchiere o di uno strumento apposito si ritagliano dei dischi ed al loro centro si depone il ripieno fatto con i gamberetti. Li si richiude avendo l'accortezza di bagnare leggermente i bordi in modo da far aderire bene i due lembi di pasta.

Preparare poi il sugo.

Sia dall'arancio che dal limone togliere delle striscioline di buccia (attenzione a rimuovere il bianco) che andranno ad essere usate come guarnizione nel piatto, quindi si spremono i succhi di entrambi gli agrumi in una bacinella, vi si aggiunge 1 cucchiaio di farina e una presa di sale e si mescola il tutto fino a togliere ogni grumo.

In una padella si mette a sciogliere una noce di burro, quando inizia a friggere si aggiunge il succo degli agrumi e si lascia condensare.

Nel mentre si mette a bollire una pentola d'acqua con del sale grosso e, quando l'acqua bolle, si immergono i ravioli. Cuocerli per 3 o 4 minuti (o di più nel caso, dipende dallo spessore della sfoglia) quindi, delicatamente, li si estrae dalla pentola e li si mette nella padella con il sugo bello caldo e li si mescola.

Servire accompagnandoli nel piatto con le scorze degli agrumi e, a piacere, con una spruzzata di aceto balsamico