

## Cenni di computabilità Applet

**Stefano Mizzaro**

Dipartimento di matematica e informatica  
Università di Udine  
<http://www.dimi.uniud.it/mizzaro/>  
mizzaro@dimi.uniud.it  
Programmazione, lezione 24  
22 febbraio 2007

## Scaletta

- Esame
- Computabilità
  - Problemi, domande, risposte
  - Algoritmi, programmi, linguaggi di programmazione
  - Algoritmi che calcolano funzioni
  - Funzioni computabili e non
- Applet
- Fine corso

2

## Esame (1/2)

- Traccia esecuzione: condizione necessaria
- Scritto + progetto (facoltativo) + orale
- Orale obbligatorio se  $>26$  o  $<21$ ; a nostra scelta negli altri casi
- Voto max. senza progetto: 27
- Progetto dà incremento di 0-3 punti
  - Solo se voto scritto  $\geq 24$ !! (era: 21)
  - Se voto  $<24$ , non consegnato, non presentato  $\Rightarrow$  progetto annullato (come non fatto)
- Se voto scritto  $\leq 10 \Rightarrow -5$  all'appello succ.!!

3

## Esame (2/2)

- Media pesata dei 2 compitini:
  - $2/3 * \text{primo} + 1/3 * \text{secondo}$
- Scelta obbligatoria fra 1o scritto e 2o compitino
  - **INDICATELO QUANDO VI ISCRIVETE SU SINDY!**
- + bonus autovalutazione (solo se comunque suff.)
  - 1a - primi 5: +2
  - 2a (?) - primi 5-10(?): +3

4

## Progetto

- Progetto di gruppo (3 persone)
- Cfr. descrizione sulle mie pag. Web
  - (o su quelle di Coppola)
  - Ca. 2 settimane prima dello scritto
  - Consegnato allo scritto E spedito per email
- Altri progetti per gli appelli successivi
- Orale collegato al progetto (non allo scritto)
  - Se consegnate il progetto e non vi presentate all'orale, il progetto viene annullato
- Dovrete vedervi per conto vostro qcosa (di mooolto semplice) delle API
- Indicazioni più precise sulla descrizione del progetto

5

## Scaletta

- Esame
- Computabilità
  - Problemi, domande, risposte
  - Algoritmi, programmi, linguaggi di programmazione
  - Algoritmi che calcolano funzioni
  - Funzioni computabili e non
- Applet
- Fine corso

6

## Cosa fa un programma?

- Fornisce **risposte** a **domande**
  - Qual è la radice quadrata di 2345?
  - Qual è il massimo comun divisore di 234 e 36?
- Fornisce risposte a una **classe di domande**
  - Programma per calcolare la radice quadrata di un numero
- "Risolve **problemi**"

Stefano Mizzaro - Computabilità + Applet

7

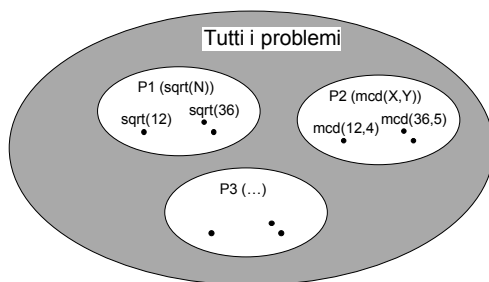
## Definizioni (1/2)

- **Problema** = insieme di domande omogenee
  - Esempi:
    - P1 = Calcolare  $\text{sqrt}(n)$
    - P2 = Calcolare  $\text{mcd}(x,y)$
- **Domanda** ("istanza di un problema")
  - Esempi:
    - I1 = Quanto vale  $\text{sqrt}(341)$ ?
    - I2 = Quanto vale  $\text{mcd}(48,36)$ ?

Stefano Mizzaro - Computabilità + Applet

8

## Problemi e domande



Stefano Mizzaro - Computabilità + Applet

9

## Definizioni (2/2)

- **Risposta** (a una domanda) = "soluzione di un'istanza"
- **Soluzione di un problema** = metodo generale che permette di fornire in modo uniforme la risposta a tutte le istanze di un problema **Algoritmo risolvete**

Stefano Mizzaro - Computabilità + Applet

10

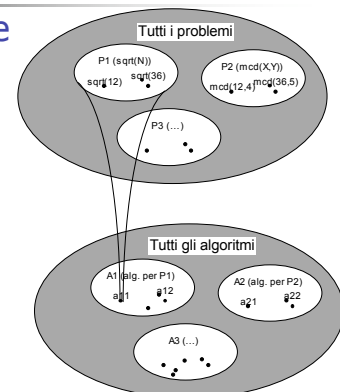
## Algoritmi e programmi

- **Algoritmo**
  - "Metodo", "procedimento", "sequenza di passi"
  - Astratto
  - Non si può toccare/vedere
- **Programma**
  - Rappresentazione precisa di un algoritmo in un certo **linguaggio di programmazione**
  - Per eseguire/comunicare un algoritmo bisogna rappresentarlo!!

Stefano Mizzaro - Computabilità + Applet

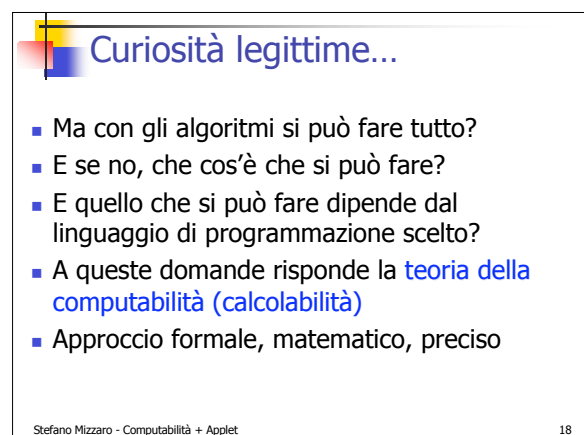
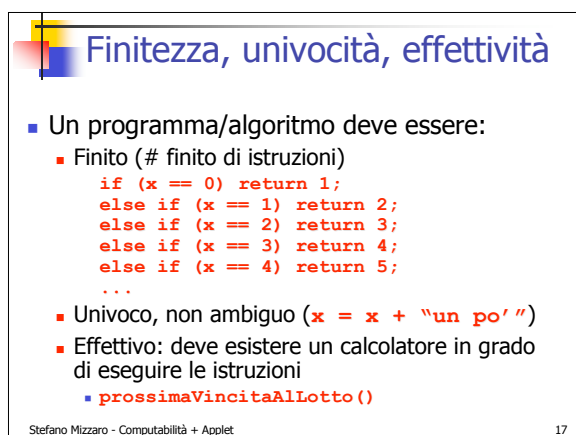
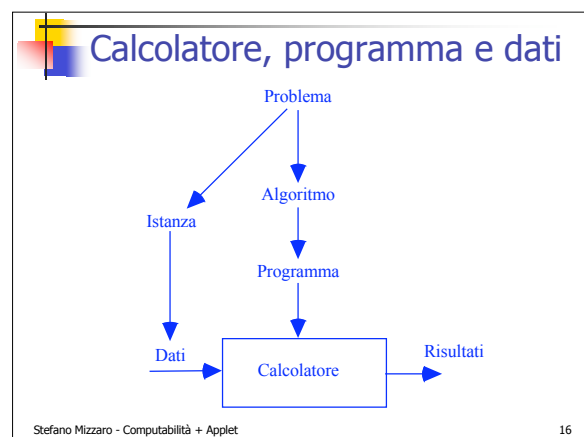
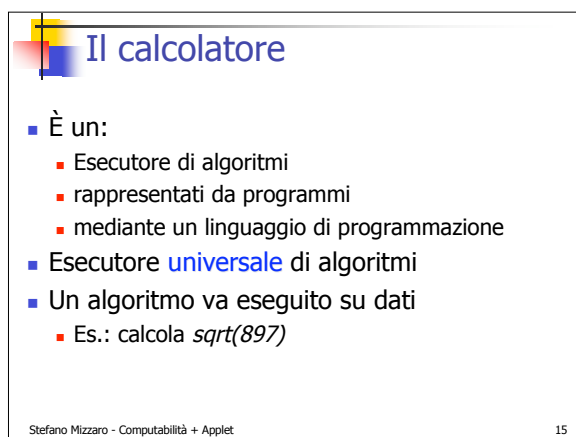
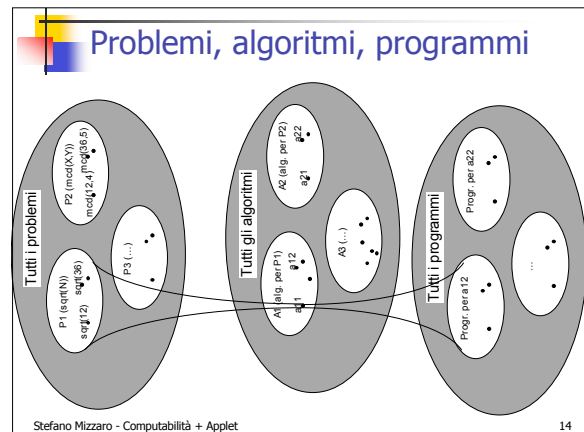
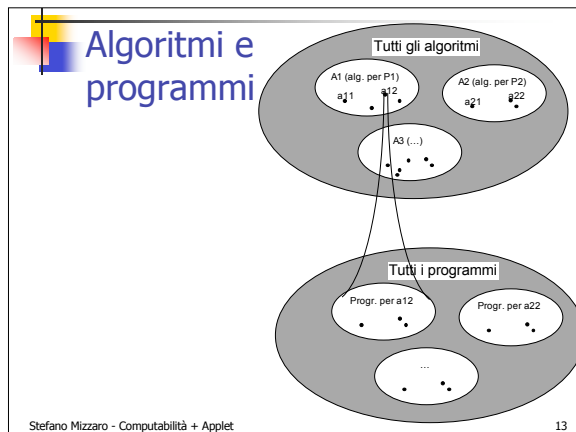
11

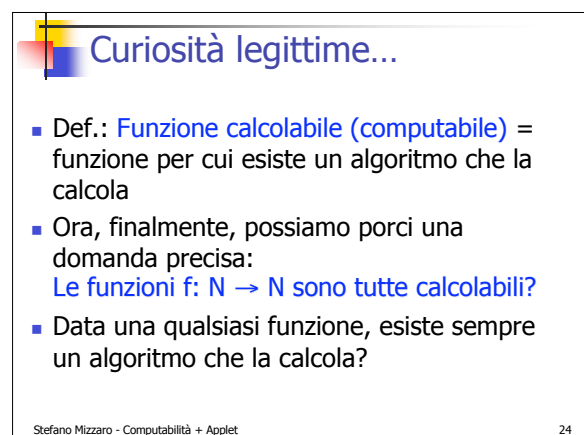
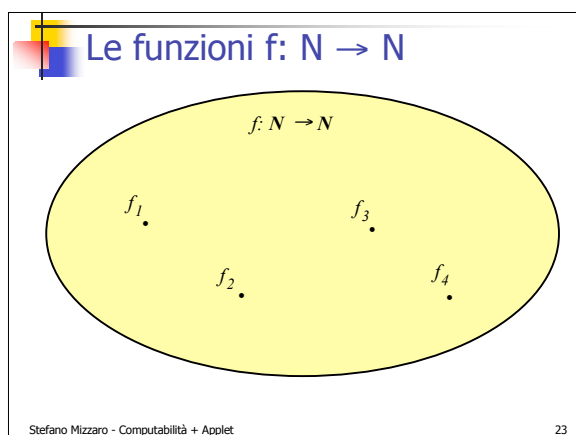
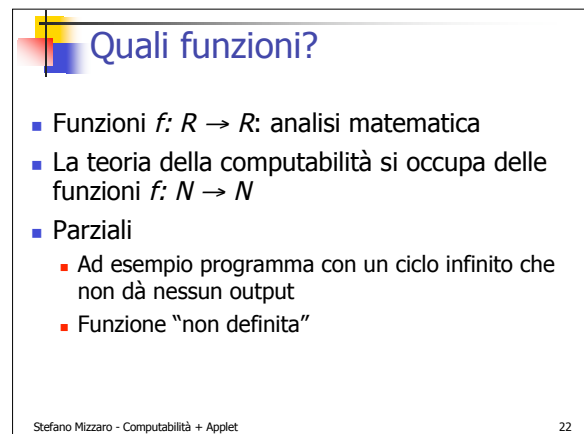
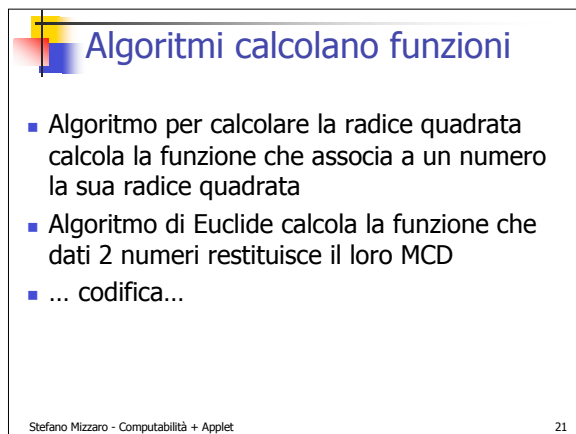
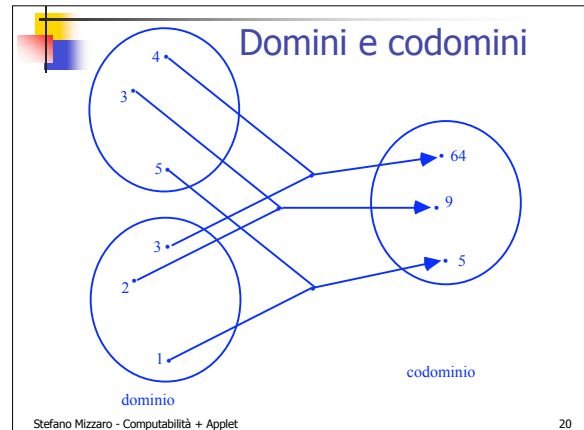
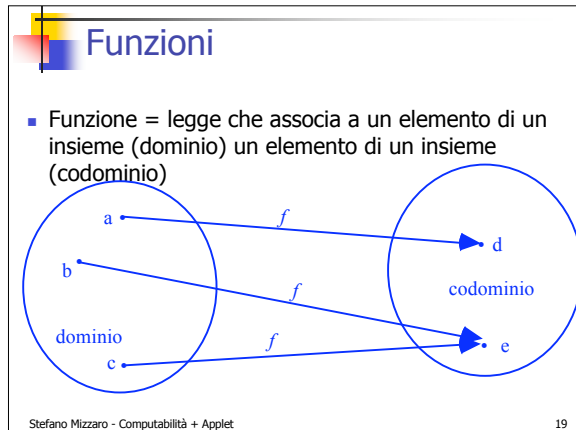
## Problemi e algoritmi

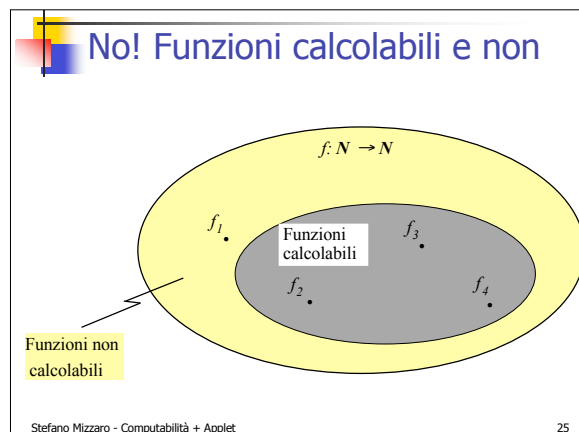


Stefano Mizzaro - Computabilità + Applet

12



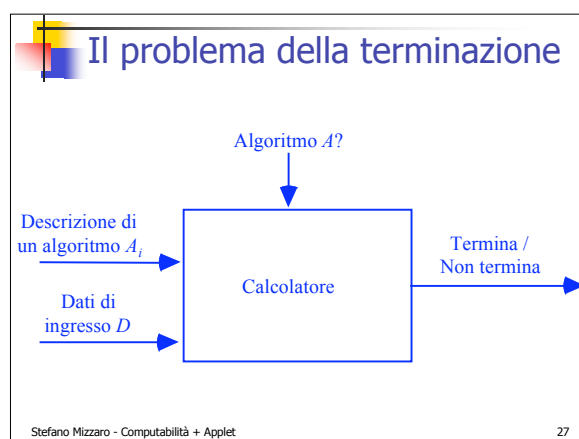




### No? Voglio un esempio...

- Il problema della terminazione (*Halting problem*)
  - Trovare un algoritmo  $A$  in grado di dirci se un qualsiasi algoritmo  $A_i$  su dati  $D$  termina.
- (un algoritmo potrebbe non terminare perché entra in un ciclo infinito)

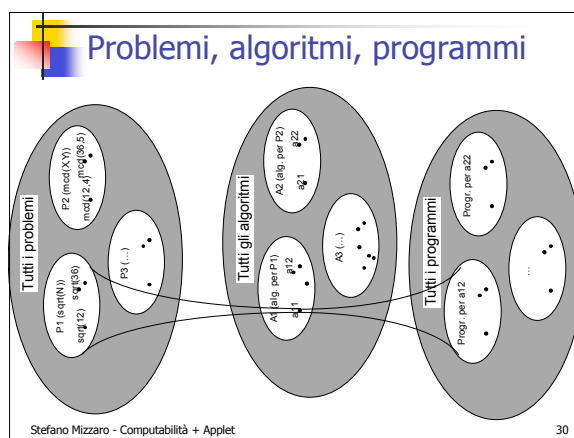
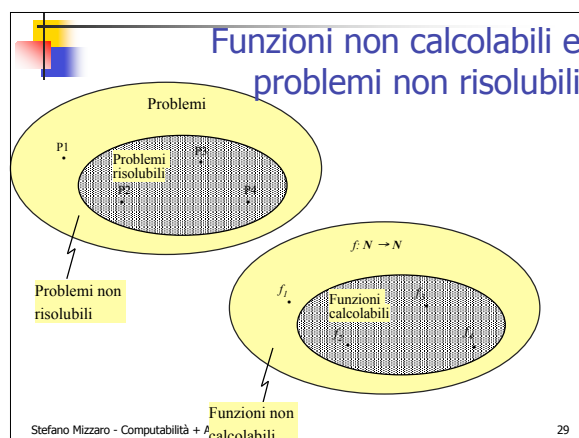
Stefano Mizzaro - Computabilità + Applet 26

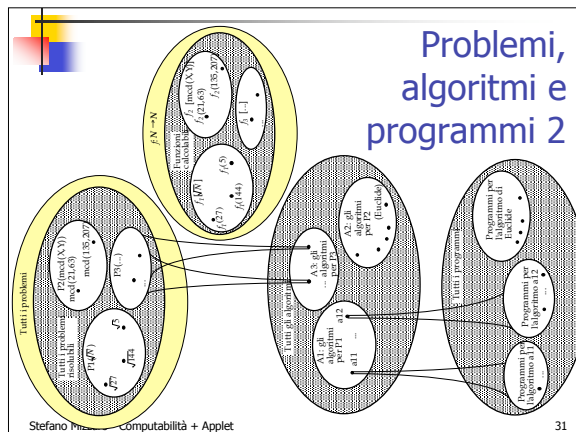


### Una soluzione?

- Beh, potrei far eseguire  $A_i$  sui dati e attendere
- Se termina so che termina
- Se dopo 2 ore non ha terminato... so che non ha terminato in due ore... ma quanto devo aspettare?!
- Non va!
- Teorema:  $A$  non esiste

Stefano Mizzaro - Computabilità + Applet 28





### Riferimenti

- <http://www.dimi.uniud.it/mizzaro/papers/algorithmo.pdf>
- Capp. 3 e 4
- Sostituite:
  - Ufficio / Calcolatore
  - Impiegato / Processore
  - Diagramma di flusso / Programma

Stefano Mizzaro - Computabilità + Applet

### Scaletta

- Esame
- Computabilità
  - Problemi, domande, risposte
  - Algoritmi, programmi, linguaggi di programmazione
  - Algoritmi che calcolano funzioni
  - Funzioni computabili e non
- Applet
- Fine corso

Stefano Mizzaro - Computabilità + Applet

### Scaletta applet

- Gli applet
  - Esempi
  - Restrizioni
- Cenni a:
  - Eventi del mouse
  - Choice
  - Menu

Stefano Mizzaro - Computabilità + Applet

### Gli applet

- Applet = "piccola applicazione"
- Non un programma indipendente, ma un programma integrato in una pagina web, in esecuzione sulla JVM del browser
- Restrizioni di sicurezza

Stefano Mizzaro - Computabilità + Applet

### La classe Applet

- `java.applet.Applet` è una sottoclasse di `java.awt.Panel`
- Da sovrascrivere per creare i nostri applet
- Non c'è un main:
  - i metodi di `Applet` vengono chiamati (implicitamente, dal browser)
  - e vanno quindi sovrascritti dal programmatore

Stefano Mizzaro - Computabilità + Applet

## Il nostro primo applet

```
import java.applet.*;
import java.awt.*;
public class CiaoATuttiApplet extends Applet {
    public void paint (Graphics g) {
        g.drawString("Ciao a tutti!", 5, 25);
    }
}
```

Stefano Mizzaro - Computabilità + Applet

37

## Il file html

```
<html>
<head>
    <title>Salve a tutti voi</title>
</head>
<body>
    <p>Il mio primo applet dice:
        <applet code="CiaoATuttiApplet.class"
            width="150" height="25">
        </applet></p>
    </body>
</html>
```

Stefano Mizzaro - Computabilità + Applet

38

## Come funziona

- Il browser ha una JVM, in grado di eseguire codice Java
- Abbiamo semplicemente sovrascritto **paint...**
- ...che viene invocato implicitamente
  - Quando c'è bisogno di disegnare l'applet

Stefano Mizzaro - Computabilità + Applet

39

## L'appletviewer

- Altro "attrezzo" del JDK
- Visualizza applet, ma non html
- Ha bisogno di un file con il tag "**applet**"
- ...a volte i browser fanno scherzi strani...

Stefano Mizzaro - Computabilità + Applet

40

## Uso dell'appletviewer

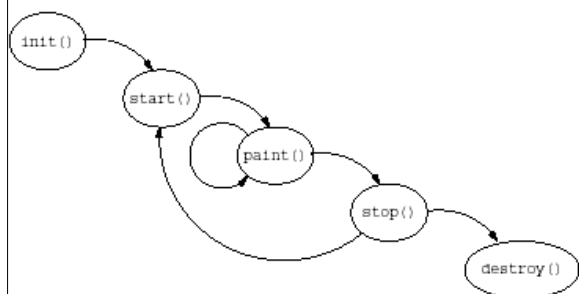
```
/*<applet code="CiaoATuttiApplet1.class"
    width="150" height="25"></applet>*/
import java.applet.*;
import java.awt.*;
public class CiaoATuttiApplet1 extends Applet {
    public void paint (Graphics g) {
        g.drawString("Ciao a tutti!", 5, 25);
        g.fillOval(0, 0, 100, 100);
    }
}
```

```
>javac CiaoATuttiApplet.java
>appletviewer CiaoATuttiApplet.java
```

Stefano Mizzaro - Computabilità + Applet

41

## "Ciclo di vita" di un applet



Stefano Mizzaro - Computabilità + Applet

42

## Metodi di applet

- Metodi (vuoti) da **Applet**:
  - init**: inizializzazione (~costruttore)
  - start**: inizio esecuzione
  - stop**: fine esecuzione, arresto
  - destroy**: rilascio risorse
  - (leggere variazioni fra i vari browser...)
- Da **Component**:
  - paint**: quello che l'applet mostra

Stefano Mizzaro - Computabilità + Applet

43

## Invocazione metodi di Applet

```

/*<applet code="MetodiApplet.class" width="10"
height="10"></applet>*/
import java.applet.Applet;
import java.awt.*;
public class MetodiApplet extends Applet {
    public void init() {
        System.err.println("init...");
    }
    public void start() {
        System.err.println("start...");
    }
    public void stop() {
        System.err.println("stop...");
    }
    public void
destroy() {System.err.println("destroy...");}
    public void paint(Graphics g) {
        System.err.println("paint...");
    }
}

```

>javac MetodiApplet.java  
>appletviewer MetodiApplet.java  
(e con un browser...)

Stefano Mizzaro - Computabilità +

44

## Applet per disegnare col mouse

- Vediamo un applet che ci consente di disegnare trascinando il mouse
- Avremo bisogno di:
  - Un applet contenente un **Canvas**
  - Ascoltatori di **eventi sul mouse** (click e drag) associati al **Canvas**
- Schema di disegno differente:
  - paint** non disegna
  - Al drag, nell'ascoltatore, disegna un segmento

Stefano Mizzaro - Computabilità + Applet

45

## ScribbleApplet1.java (1/2)

```

/*<applet code="ScribbleApplet1.class"
width="500" height="400"></applet>*/
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;

public class ScribbleApplet1 extends Applet {
    private Canvas c = new MyCanvas();
    public void init() {
        this.setLayout(new BorderLayout());
        this.add(c, BorderLayout.CENTER);
    }
}

```

Stefano Mizzaro - Computabilità + Applet

46

## ScribbleApplet1.java (2/2)

```

class MyCanvas extends Canvas {
    private int lastX, lastY;
    public MyCanvas() {
        addMouseListener(new AscoltaClick());
        addMouseMotionListener(new AscoltaDrag());
    }
    class AscoltaClick extends MouseAdapter {
        public void mousePressed(MouseEvent e) {
            lastX = e.getX(); lastY = e.getY();
        }
    }
    class AscoltaDrag extends MouseMotionAdapter {
        public void mouseDragged(MouseEvent e) {
            int x = e.getX(); int y = e.getY();
            getGraphics().drawLine(lastX, lastY, x, y);
            lastX = x; lastY = y;
        }
    }
}

```

>javac ScribbleApplet1.java  
>appletviewer ScribbleApplet1.java

Stefano Mizzaro - Computabilità

## Eventi del mouse (1/2)

- Non rispettano la regola dell'AWT "1 evento ↔ 1 ascoltatore"
- Una classe per gli eventi:
  - MouseEvent**
- Due interfacce per gli ascoltatori:
  - MouseListener** e **MouseMotionListener**
- Ci sono anche gli adapter:
  - MouseAdapter** e **MouseMotionAdapter**

Stefano Mizzaro - Computabilità + Applet

48



## Eventi del mouse (2/2)

- **MouseListener**
  - Pulsante mouse premuto (`mousePressed`)
  - Pulsante mouse rilasciato (`mouseReleased`)
  - Click (premuto + rilasciato) (`mouseClicked`)
  - Cursore entra nel componente (`mouseEntered`)
  - Cursore lascia il componente (`mouseExited`)
- **MouseMotionListener**
  - Mouse mosso (`mouseMoved`)
  - Mouse trascinato (`mouseDragged`)
- **Component:**
  - `addMouseListener` e `addMouseMotionListener`

Stefano Mizzaro - Computabilità + Applet

49

## Aggiungiamo i colori

- Scelta del "colore della penna"
- Potremmo farlo con dei pulsanti, 1 per colore
- Ma non è il componente giusto
- Usiamo invece `java.awt.Choice`

Stefano Mizzaro - Computabilità + Applet

50

## ScribbleAppletColor (1/2)

```

/*<applet code="ScribbleAppletColor.class"
   width="500" height="400"></applet>*/
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
public class ScribbleAppletColor extends Applet {
    private Canvas c = new MyCanvas();
    private Choice ch = new Choice();
    private static final Color[] colors =
        {Color.BLACK, Color.RED, Color.GREEN, Color.BLUE};
    private static final String[] colorNames =
        {"Nero", "Rosso", "Verde", "Blu"};
    public void init(){
        this.setLayout(new BorderLayout());
        this.add(c, BorderLayout.CENTER);
        for (int i = 0; i < colorNames.length; i++)
            ch.add(colorNames[i]);
        ch.addItemListener(new AscoltaColore());
        this.add(ch, BorderLayout.EAST);
    }
}

```

Stefano Mizzaro - Computabilità + Applet

51

## ScribbleAppletColor (2/2)

```

class AscoltaColore implements ItemListener {
    public void itemStateChanged(ItemEvent e) {
        c.setForeground(
            colors[ch.getSelectedIndex()]);
    }
}

class MyCanvas extends Canvas {
    // idem...
}

```

Stefano Mizzaro - Computabilità + Applet

52

## Choice

- "Scelta" di un'alternativa da un elenco
- Metodi (vedi documentazione API):
  - Costruttore
  - `add(String)`
  - `addItemListener(ItemListener)`
  - `int getSelectedIndex()`
  - `String getSelectedItem()`
- **ItemEvent:** alla scelta
- **ItemListener**
  - `itemStateChanged(ItemEvent)`

Stefano Mizzaro - Computabilità + Applet

53

## Restrizioni sugli applet

- Un applet non può:
  - accedere al file system del client
  - aprire connessioni di rete con un calcolatore che non sia il web server
  - stampare
  - creare processi figli
- Tutte le finestre aperte da un applet riportano l'indicazione che sono state create da un applet (esempio →)

Stefano Mizzaro - Computabilità + Applet

54

### Applet che apre un Frame

```

/*<applet code="OpenFrameApplet.class"
width="10" height="10"></applet>*/
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
public class OpenFrameApplet extends Applet {
    public void init() {
        final Frame f = new Frame();
        MenuBar mb = new MenuBar();
        Menu m = new Menu("File");
        MenuItem mi = new MenuItem("Esci");
        m.add(mi);
        mb.add(m);
        f.setMenuBar(mb);
        f.setVisible(true); f.pack();
        mi.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                f.dispose();
            }
        });
    }
}

```

>javac OpenFrameApplet.java  
 >appletviewer OpenFrameApplet.java

Stefano Mizzaro - Computabilità + Applet

### Commenti

- **MenuItem, Menu, MenuBar, add, setMenuBar** (vedi documentazione API)
- Menu solo nei **Frame**
- Selezione voce da menu:
  - **ActionEvent** → **ActionListener** → **actionPerformed**
- **final Frame f** perché ascoltatore con classe anonima

Stefano Mizzaro - Computabilità + Applet

### Riassunto applet

- Gli applet
  - Esempi
  - Restrizioni
- Eventi del mouse (cenni)
- **Choice**
- Menu (cenni)

Stefano Mizzaro - Computabilità + Applet

### Riassunto del corso

- Introduzione alla programmazione
  - Accento sulla qualità
- In dettaglio
  - La programmazione strutturata
  - Il linguaggio Java
- Solo accennati
  - Programmazione OO
  - API
- Cenni computabilità

Stefano Mizzaro - Computabilità + Applet