Massimo Benerecetti

Analisi Probabilistica degli Algoritmi

Lezione n.#

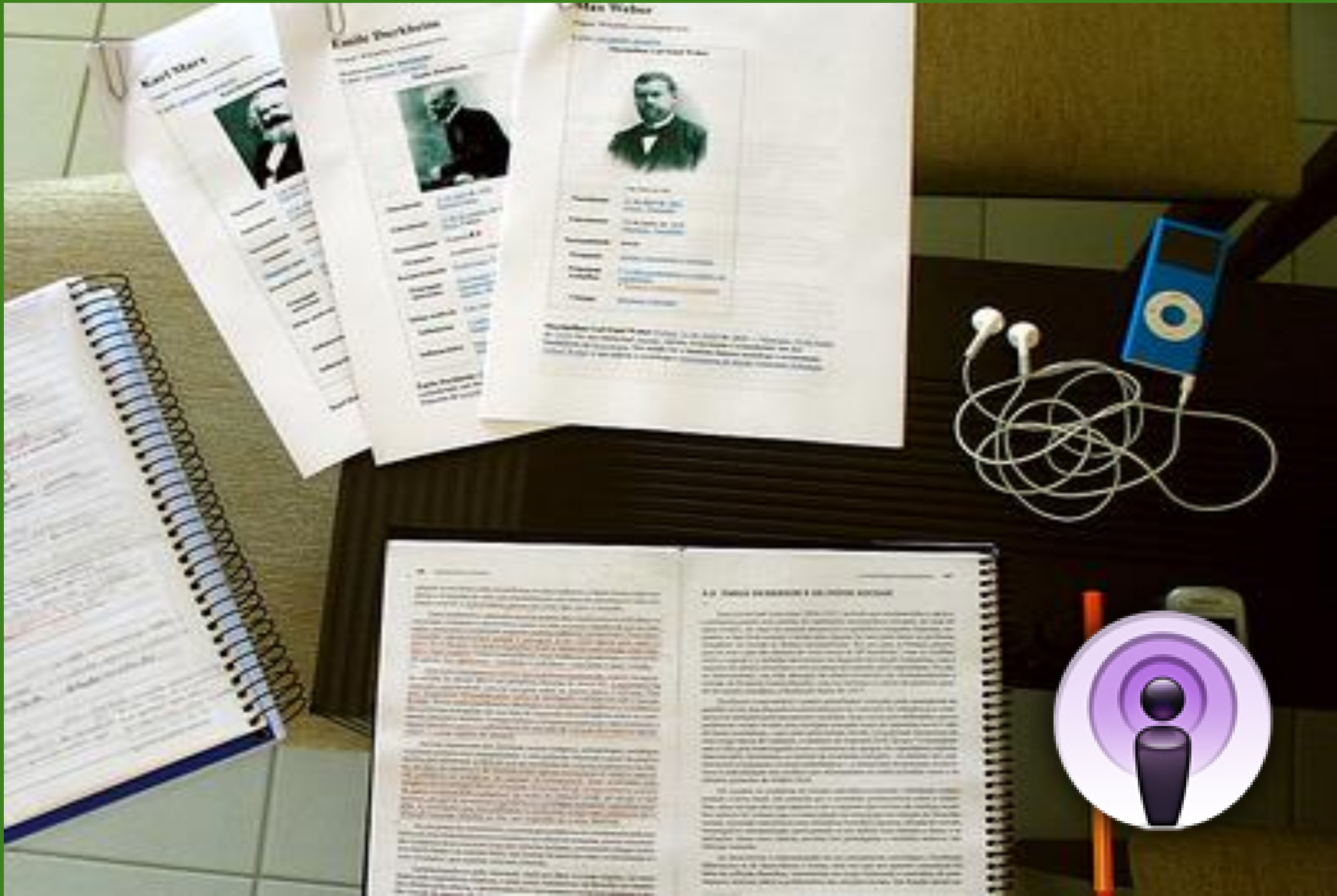
Parole chiave:
Inserire testo

Corso di Laurea:
Informatica

Insegnamento:
Algoritmi e Strutture Dati

Email Docente:
bene@na.infn.it

A.A. 2009-2010



- Le **probabilità** sono definite su uno ***spazio di campioni S***, i cui elementi sono detti ***eventi elementari***.
- Distribuzione di Probabilità è ***uniforme*** se tutti gli eventi elementari sono equiprobabili. Quindi se **n** sono egli eventi elementari, la **probabilità** di ciascun evento è pari a **1/n**.
- Un ***evento*** è un qualsiasi ***sottoinsieme dello spazio dei campioni***.
- **S** è detto ***evento certo***, $\emptyset$ è detto ***evento nullo*** (o **impossibile**)
- Siano **A** e **B** due eventi qualsiasi (sottoinsiemi di ***S***)
  - $0 \leq P(A) \leq 1$
  - $P(S) = 1$ e $P(\emptyset) = 0$
  - $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
- Dato un evento **A**, indicheremo con $\bar{A} = S \setminus A$ l'evento complemento di **A**.
  - $P(\bar{A}) = P(S) - P(A) = 1 - P(A)$

- Due eventi **A** e **B** sono detti ***mutuamente esclusivi*** (o *incompatibili*) se vale che $A \cap B = \emptyset$. In questo caso:
  - $P(A \cup B) = P(A) + P(B)$ $[P(A \cap B) = P(\emptyset) = 0]$
- ***Probabilità Condizionata***: la probabilità che si verifichi **A** dato che si è verificato **B**.
  - $P(A|B) = P(A \cap B) / P(B)$
- Due eventi **A** e **B** si dicono ***indipendenti*** se
  - $P(A|B) = P(A)$
  - $P(B|A) = P(B)$
- Segue che se **A** e **B** sono **indipendenti**, allora
  - $P(A \cap B) = P(A) \cdot P(B)$

Supponiamo di effettuare il lancio di una moneta. Lo spazio dei campioni sarà **S** = **{T,C}** (testa, croce). Assumendo che la moneta non sia truccata, possiamo considerare gli *eventi elementari* come ***equiprobabili***. Allora:

– **P(T)** = **P(C)** = **½**

Supponiamo ora di effettuare il lancio di due monete. Lo spazio dei campioni sarà quindi **S** = **{TT,TC,CT,CC}**.

- I lanci delle due monete possono considerarsi indipendenti, quindi la probabilità di ottenere due teste è:

$$P(TT) = 1/4 \; [\text{inoltre, } P(TC) = P(CT) = P(CC) = 1/4]$$

- La probabilità di ottenere almeno una testa è:

$$P(TT \cup TC \cup CT) = P(TT) + P(TC) + P(CT) = 3/4$$

$$P(TT \cup TC \cup CT) = P(\overline{CC}) = P(S) - P(CC) = 1 - 1/4 = 3/4$$

Supponiamo di effettuare il lancio di due monete e di sapere che almeno una delle due dà come risultato testa. Qual è, allora, la probabilità che entrambi i lanci abbiano dato testa?

- Chiamiamo **A** l'evento "entrambi i lanci danno testa" ({**TT**}) e **B** "almeno un lancio dà testa" ({**TT**, **TC**, **CT**}).
- Chiaramente **A** non è indipendente da **B**. Infatti:
  - la probabilità di ottenere due teste sapendo che almeno un lancio dà testa è:

$$P(A|B) = P(A \cap B) \,/\, P(B) = 1/4 \,/\, 3/4 = 1/3 \neq 1/4 = P(A)$$

si noti, infatti, che in questo caso l'evento

$$A \cap B = \{TT\} \cap \{TT, TC, CT\} = \{TT\} = A,$$

quindi $P(A \cap B) = P(A) = 1/4$.

Sia data un'urna con 4 palline indistinguibili al tatto, 2 nere e 2 bianche. Si considerino 2 *modalità di estrazione* di 2 palline dall'urna:

1. estrazione delle palline in blocco;
2. estrazione delle palline con rigenerazione (ogni pallina estratta viene rimessa nell'urna prima dell'estrazione successiva).

Qual è, nelle due modalità, la probabilità di estrarre **2 palline bianche**?

- Lo spazio dei campioni è ***S* =** {bb,bn,nb,nn}.
- Chiamiamo **A** l'evento "la prima pallina è bianca" ({bb,bn}) e **B** "la seconda pallina è bianca" ({bb,nb}) .
- In entrambi i casi (1 e 2), **P(A) =** $^{2}/_{4}$ **= ½**.

- In entrambi i casi (1 e 2), **$P(A) = {}^{2}/_{4} = {}^{1}/_{2}$**.
- Nella prima modalità, **B** non è indipendente da **A**. Dopo aver estratto la prima pallina bianca (evento **A**) nell'urna restano *3 palline*: 1 pallina bianca e 2 pelline nere.
- Quindi **$P(B|A)$** $= {}^{1}/_{3}$
- Otteniamo allora che nella prima modalità:
  - la probabilità di ottenere due palline bianche è:

$$P(A \cap B) = P(A) \cdot P(B|A) = {}^{2}/_{4} \cdot {}^{1}/_{3} = {}^{1}/_{6}$$

- Nella seconda modalità gli eventi **A** e **B** sono invece indipendenti, quindi avremo:

$$P(B|A) = P(B) = {}^{1}/_{2}$$

e, di conseguenza:

$$P(A \cap B) = P(A) \cdot P(B|A) = P(A) \cdot P(B) = {}^{1}/_{2} \cdot {}^{1}/_{2} = {}^{1}/_{4}$$

- Data una collezione di eventi $A_1, A_2, \ldots, A_n$, essi sono detti ***indipendenti a coppie*** se vale:
  - $P(A_i \cap A_j) = P(A_i) \cdot P(A_j)$ per ogni $1 \leq i < j \leq n$.
- Gli eventi $A_1, \ldots, A_n$ si dicono ***mutuamente indipendenti*** se per ogni sottoinsieme $A_{i_1}, A_{i_2}, \ldots, A_{i_k}$ (con $k \geq 2$) vale:
  - $P(A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_k}) = P(A_{i_1}) \cdot P(A_{i_2}) \cdot \ldots \cdot P(A_{i_k})$
- Supponiamo di lanciare due monete perfette. Sia A_1 l'evento "moneta 1 dà testa" A_2 l'evento "moneta 2 dà testa" e A_3 l'evento "le due monete danno risultati differenti". Allora:
  - $P(A_1) = P(\{TT, TC\}) = 1/2$
  - $P(A_2) = P(\{TT, CT\}) = 1/2$
  - $P(A_3) = P(\{TC, CT\}) = 1/2$

- Supponiamo di lanciare due monete perfette. Sia A_1 l'evento "moneta 1 dà testa" A_2 l'evento "moneta 2 dà testa" e A_3 l'evento "le due monete danno risultati differenti". Allora:
  - $P(A_1) = P(A_2) = P(A_3) = \frac{1}{2}$
  - $P(A_1 \cap A_2) = P(\{TT\}) = \frac{1}{4} = P(A_1) \cdot P(A_2)$
  - $P(A_1 \cap A_3) = P(\{TC\}) = \frac{1}{4} = P(A_1) \cdot P(A_3)$
  - $P(A_2 \cap A_3) = P(\{CT\}) = \frac{1}{4} = P(A_2) \cdot P(A_3)$
  - $P(A_1 \cap A_2 \cap A_3) = P(\emptyset) = 0 \neq 1/8 = P(A_1) \cdot P(A_2) \cdot P(A_3)$

Gli eventi A_1, A_2, A_3 sono, infatti, *indipendenti a coppie* ma *non mutuamente indipendenti*.

- Una ***variabile casuale*** è una funzione **X** che associa un numero reale **x** ad un evento. L'evento in questione è denotato con:
  - "**X**=**x**" (la variabile casuale **X** assume il valore **x**)
- ***Valore atteso*** (o valore medio)
  - $\mathbf{E[X]} = \sum_x \mathbf{x \cdot P(X=x)}$ (per probabilità discrete)
- Linearità del valore atteso
  - $\mathbf{E[X+Y]} = \mathbf{E[X]+E[Y]}$
- ***Varianza*** (dispersione delle probabilità attorno al valore atteso)
  - $\mathbf{Var[X]} = \sum_x \mathbf{(x - E[X])^2 \cdot P(X=x)} =$

    $= \mathbf{E[(X - E[X])^2]} =$

    $= \mathbf{E[X^2] - E^2[X]}$
  - Se **X** e **Y** sono **indipendenti** allora
    - $\mathbf{Var[X+Y]} = \mathbf{Var[X] + Var[Y]}$ (linearità)

Vengono lanciati due dadi. Sia X_1 la variabile causale che indica il valore del primo dado e X_2 quella per il secondo dado.

- Qual è il valore atteso della somma dei due dadi?
  - $E[X_1]=E[X_2]=[6\cdot 1/6+5\cdot 1/6+4\cdot 1/6+3\cdot 1/6+2\cdot 1/6+1\cdot 1/6]=21/6=3.5$
  - $E[X_1 + X_2] = E[X_1]+E[X_2] = 3.5+3.5 = 7$
- Qual è il valore atteso del massimo tra i valori dei due dadi (rappresentato dalla variabile X)?
  - 36 possibili casi: {(1,1),(1,2),…,(6,5),(6,1)} tutti equiprobabili
  - N. di coppie con "6" = 11 (probabilità 11/36)
  - N. coppie con "5" ma senza 6 = 9 (probabilità 9/36)
  - N. coppie con "4" ma senza 5 o 6 = 7 (probabilità 7/36)
  - N. coppie con "3" ma senza 4,5, o 6 = 5 (probabilità 5/36)
  - N. coppie con "2" ma senza 3,4,5 o 6 = 3 (probabilità 3/36)
  - N. coppie con solo "1" = 1 (probabilità 1/36).
  - Quindi $E[X] = [6\cdot 11 + 5\cdot 9 + 4\cdot 7 + 3\cdot 5 + 2\cdot 3 + 1]/36 = 4.47$

Un array $A[1..n]$ contiene n numeri distinti ordinati a caso, e ciascuna permutazione è equiprobabile.

- Qual è il valore atteso dell'indice dell'elemento massimo?
  - Sia X la variabile casuale che indica la posizione del massimo, che è associata agli eventi "$X=i$" ($A[i]$ contiene il massimo), per $1 \leq i \leq n$.
  - Essendo la permutazione causale $P[X=i] = P[X=j]$, per ogni $i \neq j$
  - $X=1, X=2, \ldots, X=n$ sono eventi mutuamente esclusivi che rappresentano tutto lo "spazio" dei campioni.
  - Avremo, quindi, che $P[X=i] = 1/n$ per ogni i.
  - Da cui, $E[X] = [1+2+3+4+\ldots+n]/n = (n+1)/2$
  - Ovvero ci si aspetta che il massimo stia nel punto di mezzo!

Se **X** e **Y** sono **indipendenti**, allora:

$$E[X \cdot Y] = E[X] \cdot E[Y]$$

Dim.: Per definizione di valore medio

$$E[X \cdot Y] = \sum_x \sum_y x \cdot y \cdot P(X=x, Y=y)$$

per l'indipendenza di **X** e **Y** avremo quindi

$$E[X \cdot Y] = \sum_x \sum_y x \cdot y \cdot P(X=x) \cdot P(Y=y).$$

Poiché x non dipende da y, avremo

$$E[X \cdot Y] = \sum_x (x \cdot P(X=x) \cdot \sum_y y \cdot P(Y=y))$$

Poiché anche y non dipende da x, avremo infine

$$E[X \cdot Y] = (\sum_x x \cdot P(X=x)) \cdot (\sum_y y \cdot P(Y=y)) = E[X] \cdot E[Y]$$

- In generale, se $X_1, X_2, \dots, X_n$ sono mutuamente indipendenti
  - $E[X_1 \cdot X_2 \cdot \dots \cdot X_n] = E[X_1] \cdot E[X_2] \cdot \dots \cdot E[X_n]$

Consideriamo l'algoritmo per trovare il massimo in una sequenza A di *n* elementi

```
int Trova-Massimo(A[])
  max = 1;
  FOR i = 2 TO n DO
    IF(A[i] > A[max]) THEN
        max = i;
  return(max)
```

- Quante volte cambia il valore di max?
- Nel caso peggiore ***O(n)***.
- E nel caso medio?

- Possiamo fare delle assunzioni sulla distribuzione di probabilità degli input.
- Su questa distribuzione possiamo calcolare i valori attesi.
- È necessario caratterizzare la distribuzione in maniera ragionevole (rispetto al problema).
- In assenza di informazioni specifiche sul problema in esame, si ricorre spesso all'assunzione di equiprobabilità dell'input.
- Per aumentare l'aderenza di questa assunzione si può tentare di forzare l'assunzione esplicitamente a livello algoritmico (**randomizzazione**).

Tecnica semplice ma potente per calcolare il valore atteso di una variabile casuale

- Dato un evento A, definiamo ***la variabile casuale indicatrice***, **I** come

$$\mathbf{I}\{A\} = 1 \quad \text{se } A \text{ si verifica}$$
$$\mathbf{I}\{A\} = 0 \quad \text{se } A \text{ non si verifica}$$

- LEMMA: Sia A un evento e $X_A = \mathbf{I}\{A\}$, allora $E[X_A] = P(A)$

Dim. Sia $\bar{A}$ il complemento di A, allora

$$E[X_A] = E[\mathbf{I}\{A\}] = 1 \cdot P(A) + 0 \cdot P(\bar{A}) = P(A)$$

Determinare il numero atteso di "testa" quando si lancia una moneta una volta

- Spazio degli eventi $\mathbf{S} = \{\mathbf{T},\mathbf{C}\}$
  - $P(\mathbf{T}) = P(\mathbf{C}) = \mathbf{1/2}$
  - $\mathbf{X_T} = \mathbf{I}\{\mathbf{T}\} = P(\mathbf{T}) = \mathbf{1/2}$

Determinare il numero atteso di "testa" quando si lancia una moneta "n" volte

- Se $\mathbf{X}$ è la variabile casuale che indica il numero di volte che esce testa in "n" lanci allora

$$E[\mathbf{X}] = \sum_{1 \le k \le n} \mathbf{k}\, P(\mathbf{X}=k)$$

- Se invece, per $1 \le \mathbf{i} \le n$, usiamo le seguenti variabili casuali indicatrici $\mathbf{X}_i = \mathbf{I}\{\mathbf{i}\text{-esimo lancio dà testa}\}$, allora

$$\mathbf{X} = \sum_{1 \le \mathbf{i} \le n} \mathbf{X}_i$$

Se usiamo variabili indicatrici $\mathbf{X}_i = \mathbf{I}\{\mathbf{i}\text{-esimo lancio dà testa}\}$ con $1 \leq \mathbf{i} \leq n$, allora

$$\mathbf{X} = \sum_{1\leq \mathbf{i} \leq n} \mathbf{X}_i$$

Poiché $E[\mathbf{X}_i] = P(\mathbf{X}_i) = \frac{1}{2}$

il numero atteso di "testa" in n lanci è quindi, per la linearità del valor medio:

$$E[\mathbf{X}] = E\left[\sum_{1\leq \mathbf{i} \leq n} \mathbf{X}_i\right] = \sum_{1\leq \mathbf{i} \leq n} E[\mathbf{X}_i] = \sum_{1\leq \mathbf{i} \leq n} \frac{1}{2} = n/2$$

```
int Trova-Massimo(A[])
   max = 1;
   FOR i = 2 TO n DO
     IF(A[i] > A[max]) THEN
        max = i;
   return(max)
```

- Assumiamo che l'array **A** sia casuale
- ***X*** sia variabile casuale che corrisponde al numero di volte in cui il valore di `max` cambia
- Definiamo le seguenti variabili casuali indicatrici:

$$X_i = \mathbf{I}\{A[i] > A[j] \text{ per } j<i\}, \text{ con } 1 \leq i \leq n$$

- In modo che sia $X = X_1 + X_2 + \ldots X_n$

```
int Trova-Massimo(A[])
    max = 1;
    FOR i = 2 TO n DO
      IF(A[i] > A[max]) THEN
          max = i;
    return(max)
```

- Quindi $X = X_1 + X_2 + \ldots X_n$ e
- $E[X_i]$ = $P(A[i] > A[j]$ per $j<i)$
- Ci serve conoscere $P(A[i] > A[j]$ per $j<i)$!!
- Assumendo **distribuzione uniforme**, la probabilità che il massimo tra i primi i elementi sia in proprio in posizione i è $1/i$, quindi

$$E[X_i] = 1/i$$

- Da cui ricaviamo che

$$E[X] = E\left[\sum_{1\leq i\leq n} X_i\right] = \sum_{1\leq i\leq n} E[X_i] = \sum_{1\leq i\leq n} 1/i = \ln n + O(1)$$

```
int Trova-Massimo(A[])
   max = 1;
   FOR i = 2 TO n DO
     IF(A[i] > A[max]) THEN
        max = i;
   return(max)
```

Corollario. Sotto l'ipotesi che l'array A sia casuale, allora il numero medio di volte in cui il valore di max cambia nell'algoritmo `Trova-Massimo` è $\Theta(\ln n)$.

- Nell'esempio precedente abbiamo dimostrato che se l'array è casuale, il numero medio di volte in cui max cambia valore è **$\Theta(\ln n)$**. Nel caso peggiore è invece **$\Theta(n)$**
- L'analisi appena eseguita si basa sull'assunzione di probabilità uniforme dell'input (**A** è un permutazione casuale)
- Nella pratica questa assunzione potrebbe non essere vera
- È comunque possibile imporla introducendo elementi di casualità nell'algoritmo
- Possiamo infatti calcolare una permutazione casuale degli elementi dell'array in input prima di eseguire l'algoritmo
- Una procedura genera una ***permutazione casuale uniforme*** se ogni permutazione degli ***n*** elementi ha la stessa probabilità di essere generata.
- Poiché ci sono ***n!*** differenti permutazioni, tale probabilità è **$1/n!$**
- È così possibile rendere il comportamento di un algoritmo ***indipendente*** dallo specifico input ricevuto.

```
Randomize(A[])
   n = length(A);
   FOR i = 1 TO n DO
      j = Random(i,n)
      "scambia A[i] con A[j]"
```

- dove `Random(a,b)` genera un numero tra a e b con probabilità:

$$P(\text{"Random(a,b) = x"}) = \frac{1}{(b-a+1)}$$

per ogni $a \le x \le b$.

Quindi $P(\text{"Random(i,n) = x"}) = \frac{1}{(n-i+1)}$ per ogni $i \le x \le n$.

LEMMA. L'algoritmo `Randomize` genera una permutazione casuale uniforme dell'array A.

Definizione Una k-permutazione di n elementi è una sequenza che contiene esattamente k degli n elementi senza ripetizioni.

Dati n elementi, ci sono esattamente **n!/(n-k)!** k-permutazioni distinte.

Infatti, il numero di combinazioni di **n** elementi presi a gruppi di **k** è dato da:

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!}$$

Vi sono, inoltre, **k!** permutazioni di ciascun gruppo, da cui si ottiene

$$k!\binom{n}{k} = k!\,\frac{n!}{k!\,(n-k)!} = \frac{n!}{(n-k)!}$$

Definizione Una k-permutazione di n elementi è una sequenza che contiene esattamente k degli n elementi senza ripetizioni. Dati n elementi, ci sono esattamente $n!/(n-k)!$ k-permutazioni distinte.

LEMMA. L'algoritmo `Randomize` genera una permutazione casuale uniforme dell'array A.

Dim. Per induzione sul numero i di passate del ciclo, dimostriamo che prima della i-esima passata, la porzione dell'array $A[1 \ldots i-1]$ contiene una $(i-1)$-permutazione di A con probabilità $(n-i+1)!/n!$.

- $i=1$: $A[1 \ldots 0]$ è vuota ed esiste una sola 0-permutazione che ha quindi probabilità 1 di essere generata. La proprietà desiderata vale in quanto per $i=1$, $(n-i+1)!/n!$ si riduce a $n!/n!=1$.
- Supponiamo che la proprietà vanga appena prima della i-esima passata, cioè che $A[1 \ldots i-1]$ contenga una $(i-1)$-permutazione di A con probabilità $(n-i+1)!/n!$.
- Vogliamo dimostrare, terminata la i-esima passata, $A[1 \ldots i]$ contiene una i-permutazione di A con probabilità $(n-i)!/n!$.

- Vogliamo dimostrare che terminata la i-esima passata, $A[1\ldots i]$ contiene una (i)-permutazione di A con probabilità $(n-i)!/n!$.
- Consideriamo una qualsiasi i-permutazione $\mathbf{x^i}=\mathbf{x_1,x_2,\ldots,x_{i-1},x_i}$ di A.
- $\mathbf{x^i}$ contiene una $(i-1)$-permutazione $\mathbf{x_1,x_2,\ldots,x_{i-1}}$ di A seguita dall'elemento $\mathbf{x_i}$.
- Per ipotesi induttiva,

$$P(A[1\ldots i-1]=\mathbf{x_1,x_2,\ldots,x_{i-1}}) = P(E_1) = (n-i+1)!/n!$$

- Sia E_2 l'evento "l'i-esima passata pone $\mathbf{x_i}$ in $A[i]$".
- Chiaramente $P(A[1\ldots i]=\mathbf{x_1,x_2,\ldots,x_i}) = P(E_1 \cap E_2)$
- Sappiamo, inoltre, che $P(E_1 \cap E_2) = P(E_2|E_1) \cdot P(E_1)$

- Sappiamo che $P(E_1 \cap E_2) = P(E_2|E_1) \cdot P(E_1)$
- $A[i]$ viene scelto da **`Random(i,n)`** in modo casuale tra gli elementi in $A[i…n]$ alla i-esima passata.
- Quindi $P(E_2|E_1) = P(A[$**`Random(i,n)`**$] = \mathbf{x_i}) = 1/(n-i+1)$.
- Avremo quindi che prima della $(i+1)$–esima passata:

$$
\begin{aligned}
P(E_1 \cap E_2) &= P(E_2|E_1) \cdot P(E_1) \\
&= 1/(n-i+1) \cdot (n-i+1)!/n! \\
&= 1/(n-i+1) \cdot ((n-i+1) \cdot (n-i)!/n!) \\
&= (n-i)!/n!
\end{aligned}
$$

- Poiché alla fine dell'algoritmo $i = n+1$, possiamo concludere che la porzione dell'array $A[1…n]$ contiene una n-permutazione di A con probabilità $(n-i+1)!/n! = (n-(n+1)+1)!/n! = 1/n!$.
- Quindi l'algoritmo **`Randomize`** genera effettivamente una permutazione casuale uniforme.

```
int Trova-Massimo-Rand(A[])
   Randomize(A)
   max = 1;
   FOR i = 2 TO n DO
     IF(A[i] > A[max]) THEN
        max = i;
   return(max)
```

Corollario. Per ogni array A, il numero medio di volte in cui il valore di max cambia nell'algoritmo `Trova-Massimo-Rand` è $\Theta(\ln n)$.

- Avete una "moneta" truccata che ritorna Testa con probabilità p e Croce con probabilità $1-p$, dove $p \neq ½$. È possibile definire una procedura che generi T (C) con probabilità **½** usando tale moneta?
- Consideriamo il seguente algoritmo

```
Moneta_Onesta()
  REPEAT
     x = Lancio_moneta_falsa()
     y = Lancio_moneta_falsa()
  UNTIL (x ≠ y)
  return x
```

  - $P(x=T) = P(x=T \text{ e } y=C) = p \cdot (1-p)$
  - $P(x=C) = P(x=C \text{ e } y=T) = (1-p) \cdot p$
  - $P(x=T) = P(x=C)$ sono uguali e la somma dà 1.
- Quindi $P(x=T) = P(x=C) =$ **½**, come desiderato.

In un ristorante, ognuno dei clienti consegna il suo cappello ad un inserviente all'ingresso. All'uscita, l'inserviente restituisce i cappelli ai clienti in ordine casuale. Qual è il numero atteso **X** di clienti che riceveranno il proprio cappello ?

- Introduciamo le seguenti variabili casuali indicatrici per $1 \leq i \leq n$.

$$\mathbf{X_i} = \mathbf{I}\{\text{cliente } i \text{ riceve il proprio cappello}\}$$

- $P[\mathbf{X_i}=1] = 1/n$ (distribuzione uniforme)
- quindi $E[\mathbf{X_i}] = \mathbf{1/n}$
- Ovviamente $\mathbf{X} = \mathbf{X_1}+\mathbf{X_2}+\ldots+\mathbf{X_n}$, avremo quindi:

$$E[\mathbf{X}] = E\left[\sum_{1 \leq i \leq n} \mathbf{X_i}\right] = \sum_{1 \leq i \leq n} E[\mathbf{X_i}] = \sum_{1 \leq i \leq n} 1/n = 1$$

Dato un array **A** di **n** interi distinti. Qual è il numero atteso di inversioni (**X**) nell'array [**(i,j)** è un'inversione se $i < j$ e $A[i]>A[j]$]?

- Introduciamo le seguenti variabili indicatrici con 2 indici

$$X_{ij} = I[A[i]>A[j] \text{ e } i < j]$$

- $P[X_{ij}=1] = ½$ (assumiamo distribuzione uniforme)
- Quindi $E[X_{ij}] = ½$
- Chiaramente $X = \sum_{1\le i\le n}\sum_{i+1\le j\le n} X_{ij}$ e quindi:

$$E[X] = E\left[\sum_{1\le i\le n}\sum_{i+1\le j\le n} X_{ij}\right]$$

$$= \sum_{1\le i\le n}\sum_{i+1\le j\le n} E[X_{ij}]$$

$$= \sum_{1\le i\le n}\sum_{i+1\le j\le n} ½ = n(n-1)/4$$