



COMUNE DI SANTA VENERINA

Città Metropolitana di Catania

CONCORSO PUBBLICO, PER TITOLI ED ESAMI, PER LA COPERTURA DI N. 1 POSTO DI ASSISTENTE SOCIALE A TEMPO INDETERMINATO E PARZIALE A 24 ORE – AREA DEI FUNZIONARI E DELLA E.Q.

VERBALE SORTEGGIO COMPONENTI COMMISSIONE GIUDICATRICE

L'anno 2026, il giorno 22 del mese di GIUGNO alle ore 10.30, in Santa Venerina, nei locali della sala consiliare del palazzo comunale di Piazza Regina Elena - 95010 Santa Venerina – sono presenti:

Segretario Generale - dott.ssa Nerina Scandura

Responsabile Settore 1- Affari Generali – dott. Giuseppe Licciardello

Verbalizzante– Sig. Arcangelo Arcidiacono

per procedere al sorteggio pubblico di 1 componente esterno della commissione di concorso di cui all'oggetto oltre cinque componenti supplenti da individuare tra i soggetti iscritti negli appositi elenchi predisposti dall'Assessorato Regionale dell'Autonomie Locali, ai sensi dell'art. 3, della L.R. 12/1991, giusto D.A. n. 17/serv.2 del 16.02.2023 del Dipartimento Autonomie Locali;

Si dà atto che l'odierna riunione è stata convocata con avviso regolarmente pubblicato in data 16/06/2026 nell'albo pretorio on line e nella sezione amministrazione trasparente del sito istituzionale.

Il Segretario Generale dà lettura:

- dell'art.6 del bando di concorso;
- del *Decreto Assessorato delle autonomie locali e della Funzione pubblica. Elenchi dei componenti delle commissioni giudicatrici di pubblici concorsi nella Regione, D.A. n. 17/serv.2 del 16.02.2023;*

A questo punto, il segretario dà atto che il sorteggio viene effettuato in modalità elettronica, utilizzando il sistema di generazione di numeri casuali in uso sul sito della Regione Emilia Romagna, reperibile al seguente link :

<http://wwwservizi.regione.emilia-romagna.it/generatore/default.aspx>, il quale garantisce la possibilità di estrarre in modo casuale una serie di numeri compresi tra un intervallo, generati da un seme generatore. In particolare, si procederà all'estrazione:

- ✓ di n. 6 numeri (di cui n. 1 componente effettivo e n. 5 supplenti), inserendo come estremi degli intervalli il numero 1 e il numero finale 20 corrispondente al numero di soggetti inseriti rispettivamente nell'elenco provinciale di Catania **categoria E1L** – tipologia professionale assistenza sociale.

Si procede all'individuazione dall'elenco da cui nominare i componenti di profilo professionale assistenza sociale (**E1L**). Si prende atto che nel suddetto elenco sono iscritti solo 20 componenti e pertanto si procederà alla nomina di n.6 componenti di cui il primo effettivo e gli altri 5 supplenti. Si precisa che ciascun numero estratto indica il numero di ordine di iscrizione nell'elenco provinciale. Si inseriscono pertanto nell'applicativo sopra descritto, la cui modalità di funzionamento è indicata nella scheda tecnica che si allega al presente verbale (all. 1), i numeri di intervallo iniziale 1 e finale 20, indicando che debbono essere estratti 6 numeri. Il seme generatore viene indicato dalla dipendente sig.ra Consoli Silvana nel n. 1.

Si dà atto che l'estrazione automatizzata individua i seguenti numeri: 11- 2-18-15-6-13, che corrispondono ai seguenti nominativi:

Componente effettivo (n.11)

- 1) Malatia Fabio nato a Palermo il 20.05.1972;

Componenti supplenti che subentrano in caso di rinuncia nel seguente ordine (2-18-15-6-13)

- 1) Grimaldi Roberto Ugo Maria nato a Enna il 18.08.1972;
- 2) Terrasini Michele nato a Palermo il 26.05.1965;
- 3) Greco Rosolino nato a Vicari (PA) il 09.01.1956;
- 4) Santonocito Samantha Desirè nata a Catania il 07.11.1978;
- 5) Salomone Roberto nato a Palermo il 13.11.1984.

Si allega copia della videata di estrazione (all. 2).

Così conclusi i lavori, si demanda all'ufficio personale il seguito degli adempimenti ai fini della nomina della Commissione.

Si chiudono le operazioni alle ore 11.00.

Il presente verbale viene redatto dal segretario generale, letto confermato e sottoscritto come segue.

IL SEGRETARIO GENERALE

IL RESP. AREA AFFARI GENERALI

IL VERBALIZZANTE



Risultato generazione

Generazione di **6** numeri **distinti** nell'intervallo [1, 20]
Parametri del generatore: *seme* = **1**, *m* = **2147483647**, *a* = **1103515245**
[Ulteriori informazioni sul tipo di generatore implementato.](#)
[Scarica l'elenco dei numeri](#)

- 11
- 2
- 18
- 15
- 6
- 13

GENERA ALTRO NUMERO

Informazioni sul generatore

salta a esito operazione

Si premette subito che un computer, in quanto macchina assolutamente priva di arbitrio (ovvero deterministica), non è in grado di generare dei numeri casuali nel senso letterale del termine. Quello che si può fare è realizzare un algoritmo che generi una sequenza di numeri avente le stesse proprietà statistiche di una sequenza di numeri casuali. Agli effetti pratici la sequenza generata ha lo stesso valore di una sequenza realmente casuale, ma il fatto che si utilizzi un algoritmo implica che è possibile ricreare la stessa sequenza fornendo all'algoritmo lo stesso valore iniziale (chiamato appunto *seme*).

Pertanto è più corretto parlare di generatori di numeri *pseudo*-casuali anziché di numeri casuali veri e propri. L'algoritmo utilizzato in questo sito è noto in letteratura e la sua "bontà", nel senso che genera una sequenza con le stesse proprietà statistiche di una sequenza casuale, è stata dimostrata, con i test statistici del caso. Si tratta di un generatore di Lehmer, ovvero sia un generatore *congruenziale moltiplicativo*. L'algoritmo, a partire dal seme identificato con X_0 , è definito in termini ricorsivi nel seguente modo:

$$X_{k+1} = (aX_k + c) \bmod m, k > 0$$

I numeri generati appartengono all'intervallo $[0, m-1]$ e

- m è un numero intero
- a e c sono numeri interi maggiori o uguali a 0 e minori di m

Di particolare importanza per la bontà della sequenza è la scelta dei valori di a , c e m . Nel caso di questo sito i valori scelti sono stati tratti dalla letteratura e sono: $m=2147483647$ ($2^{31}-1$), $a=1103515245$ e $c=0$.

L'implementazione deve gestire i casi di *overflow* che possono capitare durante la moltiplicazione aX_k . Dato che dopo la moltiplicazione si esegue una operazione di modulo l'*overflow* può essere evitato distribuendo la moltiplicazione su più parti e eliminando i termini più significativi (ovvero le cifre più a sinistra). Per fare ciò si può usare l'algoritmo di Schrage (vedi listato). Ecco un estratto della classe, in linguaggio C++, che implementa il generatore presente su questo sito (si noti che la sequenza genera numeri reali tra 0 e 1 (1 escluso) in quanto divide il valore generato per m):

```
class CLCG {
public:
    long Seed, M, A;

    CLCG()
    {
        Seed = 1;
        M = 2147483647; // 2^31-1
        A = 1103515245;
    }

    double GetNextValue()
    {
        // ALGORITMO DI SCHRAGE
        long Q, Z, lo, hi, test;
        Q = M / A;
        Z = M % A;
        hi = Seed / Q;
```

```

        lo = Seed % Q;
        test = A*lo - (long)(Z*hi);
        Seed = (test > 0) ? test : test + M;
        return (double) Seed / M;
    }
};

```

e qui di seguito un breve programma che usa la classe precedente e con cui si possono riprodurre le sequenze di numeri generate da questo sito:

```

void main(void) {
    CLCG LCG;
    long min, max, n;
    cout << "Min? "; cin >> min;
    cout << "Max? "; cin >> max;
    cout << "Seme? "; cin >> LCG.Seed;

    cout << "Quanti numeri? "; cin >> n;
    cout << "ATTENZIONE: non si eliminano i duplicati" << endl;

    while (n-- > 0)
        cout << long(min + LCG.GetNextValue() * (max-min)
+ 0.5) << endl;
}

```