

Compiti per le vacanze estive, classe 3 ITIA

INFORMATICA

Per studenti con giudizio sospeso o aiuto nella materia di informatica:

- Organizzare un quaderno con tutti gli schemi degli argomenti trattati e brevi esempi di codice applicativo
- Svolgere tutti gli esercizi di potenziamento allegati sotto
- Svolgere i 10 esercizi assegnati a tutta la classe

Il giorno dell'esame, consegnare il quaderno e gli esercizi (su carta o su chiavetta).

Per tutta la classe:

- Risolvere, utilizzando le funzioni, i 10 esercizi per tutta la classe allegati sotto.

Esercizi per tutta la classe (da risolvere utilizzando le funzioni)

Esercizio 1:

L'accrescimento della popolazione umana è guidato da una semplice legge matematica: l'incremento della popolazione tra un anno e il successivo è dato dal prodotto di una costante (il tasso di accrescimento) per la dimensione della popolazione. Realizza l'algoritmo che consenta di simulare anno per anno i valori della dimensione della popolazione a partire da:

- l'anno iniziale della simulazione
- il valore della dimensione iniziale della popolazione
- il valore del tasso di accrescimento
- l'anno finale della simulazione

Esercizio 2:

Scrivi un algoritmo che legge un numero e visualizza la tabellina pitagorica fino al numero inserito

Esercizio 3:

Il filosofo Zenone di Elea motivava il fatto che il moto è solo un'illusione con la seguente argomentazione: "dovendo percorrere una certa distanza si dovrà coprire con un primo spostamento metà della distanza, con un secondo spostamento metà della distanza rimanente, con un terzo spostamento metà della distanza ancora rimanente e così via senza arrivare mai a destinazione".^[1]Progetta un algoritmo che, data la distanza da percorrere e il numero di spostamenti effettuati, calcoli la distanza effettivamente coperta

Esercizio 4:

in una acciaieria il semilavorato metallico grezzo viene prodotto con uno spessore di alcuni centimetri e viene successivamente lavorato passando per una serie di n laminatoi, ciascuno dei quali diminuisce lo spessore del 10%.^[1]Progetta un algoritmo per determinare lo spessore del laminato a partire dallo spessore del semilavorato grezzo e dal numero di laminatoi presenti nel processo di lavorazione.^[1]Modifica l'algoritmo precedente in modo che determini il numero di laminatoi necessari nel processo di lavorazione per ottenere un laminato di spessore definito a partire dallo spessore del semilavorato

Esercizio 5:

Una leggenda orientale narra di un matematico che, in cambio di alcuni servigi al re, chiese la seguente ricompensa: "un chicco di riso per la prima casella di una scacchiera, due chicchi di riso per la seconda casella di una scacchiera, quattro chicchi di riso per la terza casella e così via per tutte le 64 caselle della scacchiera". Progetta un algoritmo che, a partire dal numero e di caselle che si intendono riempire, calcoli il numero complessivo di chicchi di riso che spettano come ricompensa

Esercizio 6:

La popolazione di un particolare batterio raddoppia ogni ora. Progetta un algoritmo che, a partire dal numero di ore trascorse e dal valore espresso in "unità di carica batterica" della consistenza iniziale della popolazione batterica, ne calcoli la consistenza finale raggiunta

Esercizio 7:

È dato un file di testo *persone.txt* che contiene i dati di una serie di persone, una persona per riga. Ogni riga contiene:

- il cognome
- uno o più spazi
- il nome
- uno o più spazi
- la data di nascita nel formato **gg/mm/aaaa**
- uno spazio
- un carattere che indica il sesso.

Scrivere un programma che, dopo aver definito una *struttura persona*:

1. contenga una funzione **lettura()** che, dato il nome del file (ed eventualmente altri parametri se opportuno), legga i dati delle persone dal file e li metta in un *array di persona* di nome **elenco**; Al termine stampare l'array.
2. contenga una funzione **compatibili()** che, date due persone, restituisca *vero* se le due persone sono compatibili (intendendo con questo che esse sono di sesso diverso e la differenza di età, riferita solo all'anno, non supera i 5 anni), o *falso* altrimenti; Al termine stampare l'array.
3. chieda all'utente, da console, cognome, nome, sesso e data di nascita e si inseriscano nel file

Esercizio 8:

Scrivere su un file di testo "fibonacci.txt" n numeri della serie di Fibonacci, con n fornito in input e compreso tra 5 e 40.

Leggere il file di testo e visualizzare la serie.

Esercizio 9:

Scrivere una funzione ricorsiva che azzeri tutti gli elementi pari, raddoppi gli elementi dispari e ritorni il minimo elemento di un array passato.

Esercizio 10:

Scrivere una procedura ricorsiva che riceva un array di caratteri e stampi la prima metà su una riga e la seconda metà su una riga diversa.

Se l'array ha un numero dispari di elementi, la lettera centrale dovrà essere stampata su una riga a parte, tra la prima metà e la seconda metà.

Esercizi di potenziamento per gli alunni con giudizio sospeso

1. Scrivi un algoritmo che legga in input un numero naturale N ed effettui la somma dei primi N numeri naturali
2. Scrivi un algoritmo che, dato un numero n , determini quante volte occorre sommare a n il valore 2 per ottenere un valore superiore a 100
3. Scrivi un algoritmo che visualizzi la sequenza dei numeri pari compresi tra i valori n ed m forniti come input
4. Scrivi un algoritmo che effettui la trasformazione di un numero n intero dalla base 10 alla base 2 producendo in output le singole cifre binarie (le cifre binarie saranno visualizzate in senso inverso rispetto alla corretta interpretazione numerica!)
5. Scrivi un algoritmo che produca come output la sequenza dei quadrati dei numeri interi compresi tra 1 ed n , con n fornito come input
6. Scrivi un algoritmo che legga un numero $N \leq 16$ e calcoli la potenza N -esima del numero 2 usando la sola operazione di moltiplicazione
7. Progetta l'algoritmo che, leggendo dall'ingresso due numeri X e Y (entrambi > 1), scriva in uscita le potenze di X per i primi Y valori (es. 2, 5: risultati 1, 2, 4, 8, 16)
8. Scrivi un algoritmo che visualizzi sullo schermo i multipli di 7 che hanno valore inferiore al numero 50
9. Scrivi un algoritmo che legga una sequenza di numeri interi terminanti con uno 0 e comunichi la loro somma
10. Scrivi un algoritmo che legga una sequenza di numeri interi terminanti con uno 0 e comunichi quale è il numero più piccolo, quale è il più grande e il valore medio dei numeri letti
11. Scrivi un algoritmo che legga un numero NUM di numeri e determini il valore maggiore e quante volte esso ricorre
12. Scrivi un algoritmo che trovi il quoziente e il resto di una divisione intera per differenze successive (supponendo cioè di non disporre dell'operatore di divisione)
13. Scrivi un algoritmo che visualizzi i primi 20 numeri interi
14. Scrivi un algoritmo che legga un numero NUM e stampi i suoi primi NUM multipli
15. Scrivi un algoritmo per visualizzare i primi NUM numeri pari o i primi NUM numeri dispari, dopo aver letto NUM da input e chiesto all'utente quale sequenza visualizzare
16. Scrivi l'algoritmo che legga i 13 simboli corrispondenti ai risultati delle partite inserite in una schedina del totocalcio, quindi calcoli e stampi il numero di occorrenze dei simboli "1", X e "2".
17. Scrivi l'algoritmo che legga i 21 valori di temperature registrate nei capoluoghi di regione italiani e al termine stampi il valore massimo e il valore minimo di temperatura registrati
18. Scrivi un algoritmo che legga una sequenza di numeri fino a che viene inserito il numero 0: quindi visualizzi sullo schermo quanti numeri pari e quanti numeri dispari sono stati inseriti

19. Scrivi un algoritmo che legga 5 numeri e conteggi quanti di questi sono pari e quanti sono i numeri dispari
20. Scrivi un algoritmo che legga un numero NUM e quindi successivamente esegua la somma di NUM numeri inseriti dall'utente
21. Scrivi un algoritmo che legga un numero NUM e visualizzi tutti i numeri pari inferiori a tale numero
22. Scrivi un algoritmo che legga un numero NUM di numeri e determini il valore più grande e i due che sono immediatamente a lui adiacenti
23. Scrivi un algoritmo che esegua la somma di tutti i numeri multipli di 5 compresi tra 10 e 100
24. Scrivi un algoritmo che legga un numero NUM indicato da un utente, quindi calcoli e stampi la somma di tutti i numeri compresi tra 0 e NUM
25. Scrivi un algoritmo che legga un numero NUM e un numero EXP e calcoli la potenza EXP-esima di un numero
26. Scrivi un algoritmo che legga un numero NUM di coppie di valori che corrispondono alle temperature minime e massime di una giornata in alcune città, il cui nome viene inserito prima delle rispettive temperature. L'algoritmo deve stampare il nome della città più fredda e della città più calda
27. Scrivi un algoritmo che legga un numero NUM di numeri e determini quelli che sono maggiori di 10 e minori di 100. Di questi calcoli la somma e stampi il risultato
28. Scrivi un algoritmo che legga un numero NUM di numeri ordinati e visualizzi quelli che appartengono ad una sequenza di almeno tre numeri consecutivi
29. Scrivi un algoritmo che somma i multipli di 5 o 7 minori di un numero n
30. Scrivi un algoritmo che stampa 10 potenze di n
31. Scrivi un algoritmo che costruisce la lista dei divisori di un numero
32. Scrivi un programma per stampare e contare i numeri primi minori di n
33. Scrivi un algoritmo che legga una sequenza di valori numerici fino alla lettura di un valore 0 e scriva quanti valori sono stati letti e la loro somma
34. Scrivi un algoritmo che, dato un valore numerico k, legga n valori e conti quanti di essi sono maggiori di k scrivendo il risultato
35. Modifica l'algoritmo dell'esercizio precedente scrivendo tre risultati: quanti sono i valori maggiori di k, quanti i valori uguali a k e quanti quelli minori
36. Scrivi un algoritmo che, dato un valore numerico k, legga n numeri e conti quanti di questi sono multipli di k scrivendo il risultato
37. Scrivi un algoritmo che legga una sequenza di valori numerici fino a che la loro somma è minore di 100 e scriva la somma ottenuta e quanti sono i valori letti
38. Scrivi un algoritmo che, leggendo n valori numerici, verifichi se essi sono forniti in ordine crescente o meno