Traccia_...zione.pdf



Pagina 3 di 11

Pagina 4 di 11

```
Domanda 1
Completo
```

Punteggio max.: 5,00

vertice j: 11-6, 4-2, 0-8, 9-3, 10-9

Sia data la seguente sequenza di coppie, dove la relazione i-j indica che il vertice i è adiacente al

Si applichi un algoritmo di on-line connectivity con quickfind. I nodi sono denominati con interi tra 0 e 11. Domanda e formato della risposta:

riportare il vettore id come seguenza di interi

dopo il passo 1 (i passi cominciano da 1)

dopo il passo 2 dopo il passo 3

Dopo il passo 1 (i passi cominciano da 1)

Riportare il vettore id come sequenza di interi

0123456789106

Dopo il passo 2

0123256789106

Dopo il passo 3

8123256789106

Punteggio max.: 5,00

Completo

Domanda 2

7/3/21 1:51 PM

Si ordini in maniera ascendente la seguente sequenza di interi mediante Shell sort

h =

con la sequenza di Knuth:

Domanda e formato della risposta: Data la dimensione del vettore da ordinare, da quale valore di h si parte?

25 3 12 37 4 82 5 10 90 0 37 65 23 42 71 44

Breve motivazione:

Riportare il vettore come sequenza di interiquando è 4-ordinato, avendo svolto i passi precedenti.

h = 13

Data la dimensione del vettore da ordinare, da quale valore di h si parte?

Breve motivazione:

Il valore successivo nella sequenza sarebbe 40, maggiore della dimensione del vettore.

In altre parole, 13 > N/3 (N=16)

passi precedenti. 4 0 5 10 23 3 12 37 25 42 37 44 90 82 71 65

Riportare il vettore come sequenza di interiquando è 4-ordinato, avendo svolto i

Domanda 3

Punteggio max.: 5,00

Completo

7/3/21 1:51 PM

typedef struct {

char name[10];

student *s, vet[10];

risposta.

risposta.

char surname[10];

float score; } student;

A) Qual è l'occupazione di memoria in Byte della variabile s? B) Qual è l'occupazione in Byte del vettore vet?

1. s++;

s=vet;

inizializzati (anche se non viene mostrato nel codice)

Dato il seguente pezzo di codice:

3. vet+=2; 4. s=*vet;

C) Dire se e quali delle seguenti righe di codice sono corrette. In caso contrario, giustificare la

Rispondere alle seguenti domande, supponendo che l'architettura su cui esegue sia a 64 bit, che

le variabili di tipo float occupino 32 bit, e che sia la variabile s che il vettore vet siano stati

La dimensione di un puntatore, ovvero la dimensione del'identificativo della cella di memoria, in quest o caso 8 Byte

A) Qual è l'occupazione di memoria in Byte della variabile s?

1640 Byte = 10*(10*8B + 10*8B + 4B)

B) Qual è l'occupazione in Byte del vettore vet?

C) Dire se e quali delle seguenti righe di codice sono corrette. In caso contrario, giustificare la

7/3/21 1:51 PM Pagina 5 di 11

gramma in esecuzione (es: s sta puntando all'ultima cella di vet)

Corretta, per la dualità vettore-puntatore equivale a s = &vet[0];

Sintatticamente corretta, dopo questa istruzione, s punterà ad un numero N di celle di memoria imm

Ci potrebbero essere problemi in caso le celle di memoria successive non fossero allocate per il pro

3. vet+=2; Corretta, sempre per la dualità vettore-puntatore equivale a vet = &vet[2];

s=vet;

1. s++;

ent

ediatamente successive a quelle

4. s=*vet; Scorretta per un problema di incompatibilità: starei infatti assegnando un valore (il valore puntato da

vet, ovvero vet[0]) ad una variabile di tipo puntatore.

che punta originariamente, con N numero di celle di memoria occupate da una variabile di tipo stud



Traccia_...zione.pdf





Domanda 4 Completo

Punteggio max.: 4,00

int strFindSimilar(char str1[], char str2[]); La funzione, date due stringhe str1 e str2, cerca in str1 la prima sotto-stringa "simile" a str2, dove

Si scriva una funzione C avente il seguente prototipo

due stringhe sono dette simili se:

hanno la stessa lunghezza il numero caratteri corrispondenti uguali è (strettamente) maggiore della metà della lunghezza

di str2 (>strlen(str2)/2)

Se tale sotto-stringa esiste, se ne ritorna l'indice del primo carattere. Se non esiste, ritorna -1. Esempio di esecuzione:

strFindSimilar("FifthOfNovember", "September") ritorna 6, visto che la sotto-stringa che inizia al carattere 6 ("fNovember") ha 5 caratteri uguali alla stringa "September"; strFindSimilar("Seventh", "September") ritorna -1;

strFindSimilar("JustForExample", "notEx") ritorna 4, visto che la sotto-stringa che inizia al

carattere 4 ("ForEx") ha 3 caratteri uguali (su 5) alla stringa "notEx".

Attenzione! Si consiglia l'uso degli spazi al posto delle tabulazioni per l'indentazione del codice, dal momento che il carattere TAB viene utilizzato per la navigazione della pagina da parte della piattaforma. Sono possibili, se necessarie o utili, altre funzioni oltre a quella/e richiesta/e

#include <string.h> int strFindSimilar(char str1[], char str2[]) {

int I, i, j, count; l = strlen(str2); if(strlen(str1) < I) return -1:

```
for(i=0; str1[i+l-1]!='\0'; i++){
             count = 0:
             for(j=0; j<1; j++) if(str1[i+j] == str2[j]) count++;
             if(count > I/2) return i;
           return -1;
7/3/21 1:51 PM
                                                                                                                 Pagina 7 di 11
```

Domanda 5

medio massimo. La funzione abbia prototipo: float matrMaxDiag(float M[DIM][DIM], int n);

Completo

Punteggio max.: 6,00

dove DIM è la dimensione della matrice, mentre n (n <= DIM) è il numero di righe e colonne effettivamente usato.

1.0 2.0 3.0 4.0 5.0 1.0 2.0 3.0 4.0 5.0 2.0 2.0 2.0 2.0 -2.0 2.0 2.0 2.0 2.0 -2.0 5.0/1 = 5.0

1.1

2.2

1.8

2.2

2.2

-2.3

3.3

-3.2

3.1

1.8/1 = 1.8 -0.1/2 = 6.4/3 =

-0.05

-4.4

4.2

4.9

2.133

Si scriva una funzione C che, data una matrice quadrata di float, trovi la diagonale avente valor

Per ogni diagonale va calcolata la media aritmetica delle sue caselle. Tra queste va calcolato e

2.2 5.5 1.1 3.3 -4.42.2 -3.2 2.2 4.2 4.2

3.1

4.9

5.0

-2.3

1.8

7/3/21 1:51 PM

Esempio di esecuzione (la printf non è richiesta, la fa il main):

ritornato il valor medio massimo.

Max diag average: 5.0 Attenzione! Si consiglia l'uso degli s del codice, dal momento che il caratte pagina da parte della piattaforma. Sono possibili, se necessarie o utili, a

float matrMaxDiag(float M[DIM][DIM], int n) {

if(M[n-1][0] > M[0][n-1]) maxAvg = M[n-1][0];

if(sum/count > maxAvg) maxAvg = sum/count;

if(sum/count > maxAvg) maxAvg = sum/count;

for(count=0; count<n+corner; count++) sum += M[count-corner][count];

for(count=0; count<n-corner; count++) sum += M[count][count+corner];

for(corner=-n+2; corner<0; corner++){

for(corner=0; corner<n-1; corner++{

int corner, count; float sum, maxAvg;

sum = 0;

sum = 0;

}

7/3/21 1:51 PM

10

9

 ∞

9

3

1

0

pagina da parte della piattaforma.

2

3

else maxAvg = M[0][n-1];

spazi al posto delle tabulazioni per l'indentazione tere TAB viene utilizzato per la navigazione della	
altre funzioni oltre a quella/e richiesta/e	

5.5

4.2

5.0

* 2.0/2 = 1.0

-10.5/3 = 3.5

* 3.8/4 = 0.95

15.5/5 = 3.1

5.9/4 = 1.475

Pagina 8 di 11

Pagina 9 di 11

```
return maxAvg;
Domanda 6
Completo
Punteggio max.: 8,00
 Si consideri una regione quadrata nel primo quadrante del piano Cartesiano, con coordinata (0,0)
 per l'angolo in basso a sinistra e (100,100) per l'angolo in alto a destra. Un file di testo contiene un
 elenco di rettangoli, aventi lati paralleli agli assi del piano cartesiano. Ogni rettangolo viene
 descritto in una riga del file attraverso le coordinate x ed y del suo vertice in basso a sinistra e del
 suo vertice in alto a destra, riportate in questo ordine usando spazi come caratteri separatori. Le
 coordinate sono numeri interi inclusi tra 0 e 100, estremi inclusi. Si scriva una funzione:
 int areaIntersection(FILE * fp);
 che riceve come parametro un puntatore al file (già aperto) e ritorna l'area totale delle regioni che
 si trovano all'intersezione tra due o più rettangoli.
 Esempio di esecuzione: (per facilitare le comprensione, si riporta una regione più piccola, con
 coordinate incluse nell'intervallo 0...10).
```

Se assumiamo che il il file di esempio contenga 1 2 4 4 sulla prima riga (rettangolo più in basso) e 3 3 5 8 nella seconda riga (rettangolo più in alto), l'area totale ritornata dalla funzione, per l'intersezione, è 1.

Attenzione! Si consiglia l'uso degli spazi al posto delle tabulazioni per l'indentazione del codice, dal momento che il carattere TAB viene utilizzato per la navigazione della

Sono possibili, se necessarie o utili, altre funzioni oltre a quella/e richiesta/e

10

7/3/21 1:51 PM Pagina 10 di 11

```
#include <stdio.h>
#define maxDim 100
int areaIntersection(FILE * fp) {
  char table[maxDim][maxDim] = {{'n'}};
  int fromX, toX, fromY, toY, x, y, counter=0;
  for(x=0; x<maxDim; x++){
     for(y=0; y<maxDim; y++) table[x][y] = 'n';
  }
  while(fscanf(fp, " %d %d %d %d", &fromX, fromY, toX, toY) == 4){
     for(x=fromX; x<toX; x++){}
       for(y=fromY; y<toY; y++){}
          if(table[x][y] == 'n') table[x][y] = 'o';
          else{
            if(table[x][y] == 'o'){
               table[x][y] = 'p';
               counter++;
```

11 of 11

return counter;