Università Ca' Foscari
Dipartimento di informatica

Programmazione part-time

Esame

Nome: _____

Matricola: _____

Andrea Marin , a.a. 2009/2010

Programmazione part-time
a.a. 2009/2010

Esame

Nome: _____
Andrea Marin

Teoria. (10 punti) Rispondere ai seguenti quesiti utilizzando eventualmente gli appositi spazi bianchi. Tempo totale a disposizione: 20 min.

(2pts) **1.** Dato il seguente codice:

2 pts

```
float b=0.0; float* c; float **d;
d = &c;
c = &b;
**d = 12;
*c = *c + **d + b;
```

qual è il contenuto della variabile `b` al termine dell'esecuzione del codice? (*Barrare la risposta esatta*)

(a) `0.0` (b) `12.0` (c) `24.0` (d) `36.0` (e) indeterminato

(3pts) **2.** Dato il seguente programma:

3 pts

```
void foo(int* a, int* b) {
    *a = 12 + *b;
}

int main(){
   int* x;
   int y = 18;
   foo(x, &y);
}
```

Si speighi l'errore e si riscriva il codice correggendolo.

(3pts) **3.** Date le seguenti dichiarazioni:

3 pts

```
int a,b;
int *c, *d;
int **e;
```

si stabilisca per le seguenti scritture:

- se identificano una variabile (possono comparire a sinistra di un assegnamento) *(in caso affermativo barrare la casella della colonna A)*
- il tipo *(specificarlo nella colonna B)*
- se é presente un errore che non consentirebbe la compilazione *(in tal caso barrare la colonna C e lasciare in bianco le colonne A e B)*

	A	B	C
`*c`			
`*c + **e`			
`*(e + a)`			
`b >= 0`			
`&(a+b)`			
`**(e+*c)`			
`c[a+b]`			

(2pts) **4.** Data la seguente funzione:

2 pts

```
#define MAXVET 10

void leggi(int* vett) {
   int i;
   for (i = MAXVET; i>0; i--)
      vett[i] = i;
}

int main(){
    int v[MAXVET];
    leggi(v);
}
```

Si spieghi l'errore e la si riscriva mantenendo inalterata la firma e correggendone l'errore.

Pratica. (23 punti) Nello svolgimento del seguente esame, il candidato crei una cartella con il proprio cognome e numero di matricola e la lettera c (e.g. Rossi887766c) inserendo all'interno i file corrispondenti agli esercizi che si intendono consegnare (e.g. Esercizio1.c). Le prime righe del file devono essere dei commenti che specifichino il vostro nome e cognome e l'esercizio a cui si riferiscono.

Tempo a disposizione: 1h e 45 min.

(6^{pts}) **1.** Scrivere una funzione C che date due stringhe restituisca una stringa che contenga tutti i caratteri della prima e quelli della seconda senza duplicazioni.

6 pts

(17^{pts}) **2.** Si vuole scrivere un programma C per la gestione di matrici quadrate sparse. Una matrice è sparsa se la maggior parte dei suoi elementi è nulla. La memorizzazione di questo tipo di matrici avveiene mediante una lista in cui ciascuna cella, per ogni elemento non nullo, contine le seguenti informazioni:

17 pts

- Riga: int
- Colonna: int
- Valore: float

dove `Valore` è una quantità diversa da 0. Ad esempio la matrice:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1.5 & 0 & 0.1 \\ 0 & 0 & 0 \end{pmatrix}$$

è memorizzata in una lista con tre celle (gli elementi non-zero) che contengono: $(1, 2, 1.0)$, $(2, 1, 1.5)$, $(2, 3, 0.1)$.

(a) (2 pts) Definire il tipo dati `t_matrice` per la memorizzazione delle posizioni delle matrici sparse.

(b) (4 pts) Scaricare nella propria cartella di lavoro il file alla locazione:
`http://www.dsi.unive.it/~marin/matrice.txt`
il quale per ogni riga contiene un elemento di una matrice 5×5 e scrivere una funzione che crei la matrice corrispondente.

(c) (2 pts) Scrivere la funzione `void stampa_matrice(t_matrice mat, int dimensione)` che stampi la matrice andando a capo correttamente ad ogni riga nuova.

(d) (4 pts) Scrivere la funzione `int dimensione_minima(t_matrice mat)` che restituisce la più piccola dimensione che deve avere mat per essere una matrice valida (Suggerimento: cercare il massimo tra i valori di righe e colonne).

(e) (5 pts) Scrivere la funzione
`t_matrice somma(t_matrice mat1, t_matrice mat2, int dimensione)`
che sommi due matrici.