

Università Ca' Foscari
Dipartimento di informatica

Programmazione part-time

Esame

Nome: _____

Matricola: _____

Andrea Marin , a.a. 2009/2010

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

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

```
int *a; int **b; int c;  
a = &c;  
*a = 12;  
c++;  
b = &a;  
*b = *b + 1;
```

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

- (a) 12 (b) 13 (c) 14 (d) 1 (e) indeterminato

2 pts

(3^{pts}) **2.** Dato il seguente sottoprogramma:

```
void foo(char* a) {  
    strcpy(a, 'Pippo');  
}  
int main(){  
    char* b;  
    foo(b);  
}
```

Riscrivere il codice correggendo l'errore.

3 pts

(3^{pts}) **3.** Date le seguenti dichiarazioni:

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

si stabilisca per le seguenti scritte:

- 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 è presente un errore che non consentirebbe la compilazione (*in tal caso barrare la colonna C e lasciare in bianco le colonne A e B*)

3 pts

	A	B	C
*c + *d			
*c + *e			
*e + 5			
**e >= 0			
*(a+b)			
a			
*c + 5			

(2^{pts}) 4. Data la seguente funzione:

```
#define MAXVET 10

void leggi(int* vett) {
    int i;
    for (i = 0; i<=MAXVET; 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.

2 pts

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 1 se una delle due è suffisso dell'altra. Le operazioni devono avvenire senza la duplicazione delle stringhe. Esempio:

```
char str1 [] = "Hello world!";
char str2 [] = "world!";
int ans1 = esercizio1(str1, str2);
int ans2 = esercizio1(str2, str1);
/*Sia ans1 che ans2 contengono 1 */
```

Ovviamente il sottoprogramma deve funzionare anche con stringhe diverse in ingresso.

- (17^{pts}) 2. Si vuole scrivere un programma C per la simulazione dei movimenti di una macchinina. Si supponga di avere un campo da gioco di 10x10. Sul campo da gioco ci sono 10 ostacoli le cui coordinate sono note. Una macchinina parte dalla posizione (5,5) e segue dei comandi {U, D, R, L} che stanno rispettivamente per Up (Su), Down (Giù), Right (Destra), Left (Sinistra). La macchinina, seguiti i comandi, vince se riesce a terminare il percorso senza aver urtato alcun ostacolo e senza essere uscita dal campo di gioco.

Le posizione corrente della macchinina deve essere memorizzata in un record (struct) Tposizione e i comandi sono memorizzati in una lista di caratteri. Gli ostacoli sono posizionati in un vettore di Tposizione.

- (a) (2 pts) Definire il tipo dati **Tposizione** per la memorizzazione delle posizioni degli ostacoli e della macchinina. Dichiarare il vettore di 10 **Tposizione** per memorizzarvi la posizione degli ostacoli. Definire il tipo dati **Tmosse** come lista di caratteri.
- (b) (4 pts) Scaricare nella propria cartella di lavoro il file alla locazione:
<http://www.dsi.unive.it/~marin/mosse.txt>
il quale per ogni riga contiene il carattere corrispondente alla mossa da fare, e.g.:

```
L
L
R
U
U
L
```

e scrivere una funzione che crei la lista di mosse a partire dalle posizioni specificate nel file.

- (c) (2 pts) Scrivere la funzione **void piazzaostacoli(Tposizione* ostacoli)** che posizioni gli ostacoli usando la funzione **rand**:

```
x = rand() * 10 + 1
```

restituisce un numero casuale tra 1 e 10.

- (d) (4 pts) Scrivere la funzione **int muovi(Tposizione* ostacoli, Tmosse lista)** che restituisce 1 se la macchinina completa il percorso senza incidenti, 0 altrimenti.
- (e) (5 pts) Scrivere la funzione **int controllapercorso(Tmosse lista)** che restituisce 1 se la macchinina, nel suo percorso, torna almeno due volte in una stessa posizione.

6 pts

17 pts