

Fondamenti di Informatica A-L – (Prof. P. Zingaretti)

PROVA SCRITTA – 13 gennaio 2003

Avvertenze:

- * Consegnare **solo fogli formato A4**
- * **Scrivere su un solo lato** (no fronte-retro)
- * In ordine di preferenza usare: 1) **inchiostro nero**; 2) matita; 3) inchiostro rosso; 4) inchiostro blu
- * **In testa a ciascun foglio** scrivere: cognome, nome, numero progressivo di pagina rispetto al totale
esempio per il secondo foglio di 3 consegnati: Giuseppe Russo 2/3
- * Mantenere sul banco il **libretto o altro documento di riconoscimento** fino a controllo avvenuto
- * Nient'altro deve trovarsi sul banco: **non è consentito consultare libri, dispense, appunti, ecc.**
- * La **correzione** di riferimento per l'autovalutazione verrà effettuata in questa stessa aula alle ore **12:45**
- * La consegna delle **fotocopie** dei compiti avverrà presso l'Istituto di Informatica al termine della correzione
- * La **prova orale** si terrà **lunedì 20 gennaio** alle ore **9:00** presso l'Istituto di Informatica



Si ricorda che chi si presenterà all'orale **DEVE** portare l'implementazione al computer della propria soluzione, eventualmente corretta, **corredata di tutto quanto necessario alla verifica** del corretto funzionamento.

1. (10 punti)

Si definisce quadtree un particolare albero di grado 4 in cui ogni nodo o non ha figli oppure ne ha esattamente 4 (non esistono nodi con 1, 2 o 3 figli). Si supponga inoltre che a ciascun nodo sia associato un valore intero positivo. Si definisca una struttura dati in C per un quadtree (2 punti). Si definisca poi una funzione C che visita in ordine anticipato (depth-first pre-order) un quadtree e per ogni nodo letto stampa a video il relativo valore intero (8 punti).

Attenzione: una visita del quadtree non in ordine anticipato sarà valutata solo 2 punti.

2. (7 punti)

Si definisca una funzione C che verifica (ritorna vero o falso) se una lista contiene due valori uguali consecutivi.

3. (7 punti)

Si supponga che le temperature minime e massime registrate in alcune città in un determinato giorno siano state memorizzate in un file binario secondo il formato: nome città sempre di 30 caratteri, temperatura minima e temperatura massima entrambe come float.

Si definisca una funzione C che per tutte (numero sconosciuto) le città presenti nel file stampi a video su una nuova riga la città e le due temperature.

Attenzione: se si ipotizza un file di testo, anziché binario, l'esercizio sarà valutato solo 2 punti.

4. (6 punti)

Si esprimano le seguenti dichiarazioni di variabili e funzioni (prototipi) in linguaggio C:

A è un array di 10 puntatori a puntatori a interi

B è un puntatore a un array di 20 stringhe

C è un array di 30 puntatori a funzioni che ritornano una stringa

1. (10 punti)

Si definisca una funzione C che visita in ordine anticipato (depth-first pre-order) un quadtree e per ogni nodo letto stampa a video il relativo valore intero.

```
typedef struct quadtree_tag {
    int value;
    struct quadtree_tag *node[4];
} quadtree;

void quadtree_depth_first_pre_order(quadtree *root_ptr);
void quadtree_depth_first_pre_order(quadtree *root_ptr) {
    int i;
    if (root_ptr!=NULL) {
        printf("%d ",root_ptr->value);
        for (i=0; i<4; i++)
            quadtree_depth_first_pre_order(root_ptr->node[i]);
    }
}
```

2. (7 punti)

Si definisca una funzione C che verifica se una lista contiene due valori uguali consecutivi.

```
Boolean check_pairs(list *ptr) {
    Boolean pair_found=FALSE;
    while (ptr!=NULL && ptr->next!=NULL && pair_found==FALSE)
        if (ptr->value==ptr->next->value) pair_found=TRUE;
        else ptr=ptr->next;
    return pair_found;
}
```

3. (7 punti)

Si definisca una funzione C che per tutte (numero sconosciuto) le città presenti nel file stampi a video su una nuova riga la città e le due temperature.

```
void stampa_temp(char *nomefile) {
    FILE *fp;
    char citta[31];
    float t[2];
    int n=3;
    if ((fp = fopen(nomefile,"rb"))==NULL) exit(1);
    citta[30]='\0'; //il fine stringa non sarà mai modificato
    do {
        n=fread(citta, 30, 1, fp); //1 solo elemento da 30 caratteri
        n+=fread(t, sizeof(float), 2, fp); //altri 2 elementi se tutto OK
        if (n==3) printf("\n%s, %6.2f, %6.2f", citta, t[0], t[1]);
    } while (n==3);
    fclose(fp);
}
```

4. (6 punti)

- A è un array di 10 puntatori a puntatori a interi
- B è un puntatore a un array di 20 stringhe
- C è un array di 30 puntatori a funzioni che ritornano una stringa

```
int **A[10];
char *(*B)[20];
char *(*C[30])();
```