

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

PROVA SCRITTA – 14 marzo 2005

### Avvertenze:

- \* Consegnare **solo fogli formato A4 scritti su un solo lato** (no fronte-retro)
- \* **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 **14:30**
- \* La consegna delle **fotocopie** dei compiti avverrà al termine della correzione
- \* La **prova orale** si terrà **lunedì 21 marzo** alle ore **9:00** presso il D.I.I.G.A.



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. (8 punti)

Si definisca una funzione  $C$  che restituisce la trasposta della matrice di  $N$  righe e  $M$  colonne ricevuta in input (trasposta = matrice con righe e colonne scambiate; esempio: la trasposta di  $A[2][3]=\{\{11, 12, 13\}\{21, 22, 23\}\}$  diventa  $AT[3][2]=\{\{11, 21\}\{12, 22\}\{13, 23\}\}$ ).

### 2. (16 punti)

Si scriva una funzione  $C$  che inserisce una stringa di lunghezza variabile su un albero binario di ricerca in modo ordinato in base al suo valore (somma dei codici ASCII dei caratteri costituenti la stringa stessa).

### 3. (6 punti)

Se  $f1$  e  $f2$  sono definite nel seguente modo:

```
int f1(int *x) { return (*x)++; }  
void f2(int *x, int **y) { *x=f1(*y); }
```

Cosa stampa il seguente frammento di codice?

```
int a=2, b=4 *p=&b; f2(&a, &p); printf("\nx=%d, y=%d", a, f1(p));
```

**Nota:** oltre al risultato riportare (magari graficamente) anche il ragionamento effettuato.

### 1. (8 punti)

Si definisca una funzione C che restituisce la trasposta della matrice di N righe e M colonne ricevuta in input (trasposta = matrice con righe e colonne scambiate; esempio: la trasposta di  $A[2][3]=\{\{11, 12, 13\}\{21, 22, 23\}\}$  diventa  $AT[3][2]=\{\{11, 21\}\{12, 22\}\{13, 23\}\}$ ).

```
int *trasposta(int *A, int N, int M){
    int i, j, *AT=(int *)malloc(N*M);
    for (i=0; i<N; i++)
        for (j=0; j<M; j++) *(AT+j*N+i)=*(A+i*M+j); //AT[j][i]=A[i][j] solo 2 punti
    return AT;
}
```

### 2. (16 punti)

Si scriva una funzione C che inserisce una stringa di lunghezza variabile su un albero binario di ricerca in modo ordinato in base al suo valore (somma dei codici ASCII dei caratteri costituenti la stringa stessa).

```
typedef struct btree_tag {
    char *str;
    struct btree_tag *left;
    struct btree_tag *right;
} btree;

int str_value(char *str) {
    int value=0, i=0;
    while (str[i]!='\0') value+=str[i++];
    return value;
}

void insert(btree **btree, char *str) {
    int i, value=str_value(str);
    while (*btree!=NULL) //posizionamento
        if (str_value((*btree)->str) > value)
            btree=&((*btree)->left);
        else btree=&((*btree)->right);
    //inserimento
    *btree=(btree *)malloc(sizeof(btree));
    (*btree)->str=(char *)malloc(sizeof(str)+1);
    for (i=0; i<=sizeof(str); i++) (*btree)->str[i]=str[i];
    //oppure strcpy((*btree)->str,str);
    (*btree)->left=NULL;
    (*btree)->right=NULL;
}
```

### 3. (6 punti)

$x=4, y=5$

