

PROVA SCRITTA DI FONDAMENTI DI INFORMATICA L-A
3 DICEMBRE 2004
COMPITO A

Esercizio 1 (punti 6)

Si scriva una funzione ricorsiva `float per(float a, float b, int k);` che calcoli il seguente valore

$$\sum_{j=1}^k (a/j - j*b)$$

Esercizio 2 (punti 6)

Dato il seguente programma C:

```
#include <stdio.h>
#define N 5

void cambio(int V[], int p, int *a)
{int i;
 *a=N;
 for(i=0 ; i<N; i=i+p)
 { V[i]=V[i]* 3;
 *a= *a + i;}
 p = p + i;
}

void main()
{ int S[N]={2,4,12,14,16};
 int i, j = 2, p;
 for (i=0; i<N; i++)
 S[i] = S[i] - 1;
 cambio(S,j, &p);
 for(i=0; i<N; i++)
 printf("%d\t", S[i]);
 printf("\n%d\n", j);
 printf("\n%d\n", p);
}
```

Cosa viene stampato dal programma? La risposta deve essere opportunamente motivata.

Si spieghi come ci si comporta per l'identificatore `p` definito sia nel main sia nella procedura.

Cosa fa la `#define N 5` all'inizio del programma.

Esercizio 3 (punti 6)

Data la struttura

```
struct punti{char squadra[30];
 int meglio;
 int ultima;}
```

scrivere una procedura che riceve in ingresso un array `C` di strutture `struct punti`, la sua dimensione `Dim` e modifica l'array scambiando i campi `meglio` con `ultima` se il primo è minore del secondo..

```
void cambia(struct punti C[], int Dim, int *Cambi);
```

Si scriva poi un possibile main che legge da utente gli elementi del vettore e stampi il vettore modificato dalla procedura, nonché il numero di cambi effettuati

Esercizio 4 (punti 6)

Data la seguente funzione ricorsiva:

```
int ricors(int x, int y){
 if (x <= y)
 { if (x == y) return y-2;
 else return x-2;}
 else return x + y + ricors(x-2,y+1);
}
```

Si dica qual è il valore restituito dalla funzione e si disegnano i record di attivazione nel caso in cui la funzione sia chiamata con i seguenti parametri attuali: `ricors(15,7)`.

Esercizio 5 (punti 6)

Si descriva la gerarchia delle memorie e per quale motivo è costruita.

Si dia la definizione di linguaggio macchina e linguaggio ad alto livello.

Cosa è un file system ?

Cosa si intende per environment e binding.

Qual è la differenza tra processo computazionale ricorsivo e iterativo.

SOLUZIONE

Esercizio 1

```
float per(float a, float b, int k)
{if (k == 1) return 1/a -1*b;
  else return k/a - k*b + per(a,b,k-1);
}
```

Esercizio 2

Risultato stampato dal programma

```
3 3 33 13 45
2
```

```
11
```

I due identificatori **p** definiti all'interno del main e della procedura appartengono a due scope di visibilità diversi. L'identificatore **p** del main rappresenta la variabile dichiarata dentro il main, mentre quello della procedura rappresenta il suo secondo parametro.

L'istruzione #define N 5 informa il preprocessore che, al momento della compilazione, ogni volta viene trovato l'identificatore N dentro il codice, va sostituito con il numero 5. Ai fini pratici, scrivere N oppure 5 nel codice è esattamente la stessa cosa.

Esercizio 3 (punti 2)

```
void cambia(struct punti C[], int Dim, int *Cambi)
{int i, temp;
 *Cambi=0;
 for (i=0; i<Dim; i++)
  if (C[i].meglio < C[i].ultima)
  {temp = C[i].meglio;
   C[i].meglio = C[i].ultima;
   C[i].ultima = temp;
   (*Cambi)++;
  }
}

void main()
{int i=0, cambi;
 struct punti V[20];

 printf("Inserire il nome della squadra (0 per terminare): ");
 scanf("%s", V[i].squadra);
 while (strcmp(V[i].squadra,"0"))
  {printf("Miglior punteggio: ");
```

```
scanf("%d", &V[i].meglio);
 printf("Punteggio ultima gara: ");
 scanf("%d", &V[i].ultima);

 printf("Inserire il nome della squadra (0 per terminare): ");
 scanf("%s", V[++i].squadra);
 }

 cambia(V, i, &cambi);

 int j;
 for (j=0; j<i; j++)
  printf("Squadra: %s\tMiglior punteggio: %d\tUltimo
  Punteggio: %d\n", V[j].squadra, V[j].meglio, V[j].ultima);

 printf("\nSono stati effettuati %d cambi.",cambi);
 }
```

Esercizio 4 (punti 8)

Il risultato ottenuto è 70

