

**Fondamenti di Informatica - Ingegneria delle Telecomunicazioni – Prof. Maristella Matera**

Appello del 27 Febbraio 2015

Il tempo massimo a disposizione per svolgere la prova è di **2h**. Non è permessa la consultazione di alcun materiale didattico ed è vietato utilizzare calcolatrici, telefoni, PC. **Il voto minimo per superare la prova è 18.**

**Esercizio 1 – Codifica binaria (3 punti)**

Siano date le seguenti coppie di numeri interi con segno espressi in base 10:  $(-20, 14)$  e  $(78, -82)$ .

Per entrambe le coppie, dopo aver codificato i numeri **in complemento a due su 8 bit**, si calcoli la somma e la differenza in binario, indicando la presenza di un eventuale overflow.

**Esercizio 2 – File e liste dinamiche (14 punti)**

Una compagnia di assicurazioni memorizza i dati sulle polizze RC-auto e sugli incidenti dei propri clienti in due file distinti.

Il file con le **informazioni sulle polizze** memorizza:

- nome del cliente
- la **targa** del veicolo (stringa di 7 caratteri)
- la classe di merito: un intero tra 0 e 10

Il file con le **informazioni sugli incidenti** memorizza:

- la **targa** del veicolo coinvolto nell'incidente
- la **data** dell'incidente (in un formato a scelta)
- un carattere ("S" o "N") che indica se il veicolo è stato responsabile dell'incidente ("S") o lo ha subito ("N").

La compagnia di assicurazione periodicamente aggiorna le classi di merito dei clienti in base alle informazioni pervenute sugli incidenti: se il veicolo è stato responsabile di un incidente, la classe di merito viene incrementata di 1 fino a un massimo di 10.

1. **(7 punti)** Si scriva una funzione in C che riceve come parametro il nome di un file che memorizza le informazioni sugli incidenti e, in base ai dati letti dal file, costruisce e restituisce all'ambiente chiamante una lista dinamica che per ogni veicolo (rappresentato dalla sola targa) memorizza il numero degli incidenti in cui il veicolo è stato responsabile.

2. **(7 punti)** Si scriva una funzione in C che riceve come parametri la lista creata al punto precedente e un file che memorizza le informazioni sulle polizze e crea un nuovo file in cui si riversano le informazioni sulle varie polizze con la classe di merito aggiornata.

**Esercizio 3 – Funzioni ricorsive (8 punti)**

1. **(4 punti)** Si scriva una funzione ricorsiva che, data una lista dinamica che memorizza valori interi, stampi i soli valori di posizione pari nella lista. Il primo elemento della lista è in posizione 1.

2. **(4 punti)** Si scriva una funzione ricorsiva in C che, dato un array di interi, calcoli e restituisca come valore di ritorno il valore massimo dell'array.

**Esercizio 4 – Processi (7 + 1 punti)**

Il programma seguente, salvato nel file foo.c, viene compilato con la seguente linea di comando:

```
gcc -o foo foo.c
```

Viene poi eseguito con la seguente linea di comando:

```
./foo X
```

dove X e' il vostro cognome.

Dire quale e' l'output del programma **(7 punti)**.

**Bonus (1 punto):** dire quali warning vengono generati dal compilatore e come evitarli.

```
#include<stdlib.h>
```

```
int foo(char *s, int i);
```

```
int main(int argc, char **argv){
```

```
    int x=foo(argv[1],0);
```

```
    printf("res=%d\n",x);
```

```
    return x;
```

```
}
```

```
int foo(char *s, int i){
```

```
    int a, b;
```

```
    pid_t pid;
```

```
    if (s[0]=='\0') return i;
```

```
    pid=fork();
```

```
    if (!pid){
```

```
        s=&s[1];
```

```
        return foo(s,i+1);
```

```
    }
```

```
    wait();
```

```
    return 0;
```

```
}
```