Esercizio 2 (8 punti)

Si consideri un file di testo, composto da 11 righe, creato per memorizzare le estrazioni del gioco del lotto in una certa data, come mostrato nell'esempio seguente:

```
BARI 26 55 30 41 15
CAGLIARI 83 44 43 29 37
FIRENZE 35 65 61 2 47
GENOVA 57 28 62 59 37
MILANO 62 42 56 12 21
NAPOLI 55 19 59 72 18
PALERMO 48 34 62 41 55
ROMA 81 53 50 27 51
TORINO 40 84 38 64 35
VENEZIA 5 28 52 3 53
NAZIONALE 39 3 84 75 26
```

Si considerino le seguenti definizioni del tipo di dato strutturato `t_ruota` e del tipo array con 11 celle `t_estrazioneSettimanale`:

```
typedef struct {
    char nomeRuota[10+1];
    int numeri[5];
} t_ruota;

typedef t_ruota t_estrazioneSettimanale[11];
```

Si scriva un sottoprogramma C avente come parametri il nome di un *file* di testo (assunto essere stato già creato, nello stesso formato mostrato sopra) e una variabile di tipo `t_estrazioneSettimanale`, con il seguente prototipo:

```
... conteggioAssenti(char *fileName, t_estrazioneSettimanale vett)
```

Il sottoprogramma deve aprire il file `fileName`, copiarne il contenuto nell'array `vett`, e restituire il conteggio dei valori (quanti sono) da 1 a 90 che **NON** compaiono nell'estrazione considerata.

Soluzione:

```
int conteggioAssenti(char nomefile[], t_estrazioneSettimanale vett) {
```

```

FILE *fp = fopen(nomefile, "r"); // apertura file in sola lettura

if ( fp == NULL ) { // controllo d'errore su fopen
    fprintf(stderr, "\n Errore nell' aprire il file!\n ");
    exit(-1);
}

int r = 0; // assumerà valore 11 al termine del ciclo
while ( feof(fp) == 0 ) {
    fscanf(fp, "%s", vett[r].nomeRuota);
    for (int j = 0; j < 5; j++)
        fscanf(fp, "%d", &vett[i].numeri[j]);
    r += 1;
}

fclose(fp);

int cont = 0;

for (int num = 1; num <= 90; num++) {

    flag = 1;

    for (r = 0; (r < 11) && (flag == 1); r++) {
        for (int j = 0; (j < 5) && (flag == 1); j++)
            if (num == vett[r].numeri[j])
                flag = 0;
    } // end for in r

    if (flag == 1)
        cont += 1;
}
return cont;
} // end conteggioAssenti

```

Esercizio 3 (5 punti)

Si scriva in C un sottoprogramma

```
...ruotePesanti(...)
```

che riceva come parametri in ingresso un valore intero chiamato `soglia` e una variabile `vett` di tipo `t_estrazioneSettimanale` (vedi def. nella traccia dell'esercizio 2) e restituisca:

- un vettore dinamico di tipo `t_ruota*` contenente la copia delle celle di `vett` in cui la somma dei numeri estratti superi il valore `soglia`.
- la lunghezza del vettore dinamico (numero di elementi presenti nel vettore).

N.B.: è richiesta la **corretta** definizione dei parametri in ingresso ed in uscita dal sottoprogramma.

Soluzione

```
t_ruota* ruotePesanti(int soglia,
                    t_estrazioneSettimanale vett,
                    int* lung) {

    int supera[11] = {0}, cont = 0;
    /*
    l'array supera[...] viene usato di seguito come array di valori 0 o 1:
    se una sua cella conterra' 0 allora si sara' verificato che la cella
    di vett[...] con lo stesso indice ha somma degli estratti che NON
    supera la soglia

    se una sua cella conterra' 1 allora si sara' verificato che la cella
    di vett[...] con lo stesso indice ha somma degli estratti che supera
    la soglia

    La variabile cont, conteggia il numero di celle non nulle dell'array
    supera[...]
    */
    for (int i = 0; i < 11; i++) {
        int sum = 0;
        for (int j = 0; j < 5; j++) {
            sum += vett[i].numeri[j];
        }
        if (sum > soglia) {
            supera[i] = 1;
            cont++;
        }
    } // end for

    *lung = cont; // valore di ritorno passato per indirizzo
    if (cont == 0) return NULL;

    t_ruota* ret = (t_ruota*) malloc(cont * sizeof(t_ruota));
    if (ret == NULL) return NULL;

    int k = 0;
    for (int i = 0; i < 11; i++) {
        if (supera[i] == 1) {
            ret[k] = vett[i];
            k++;
        }
    } // end for

    return ret;
} // end ruotePesanti
```