

```

1  /*****
2  Compito del 11/04/2014 - Testo di R.Giannantonio soluzione di M.G. Maffucci
3  Realizzare un programma che legge n numeri interi e li memorizza in un vettore
4  (NMAX=100); quindi, costruisce altri due vettori vettpari e vettdispari, di
5  dimensioni opportune contenenti rispettivamente gli elementi pari e gli
6  elementi dispari.
7  Su richiesta dell'utente si deve:
8  1. effettuare un nuovo/altro inserimento;
9  2. visualizzare la media;
10 3. visualizzare il vettore vettpari;
11 4. visualizzare il vettore vettdispari;
12 5. visualizzare la media dei pari;
13 6. visualizzare la media dei dispari;
14 7. terminare il programma.
15 *****/
16
17 #include <stdio.h>
18 #include <stdlib.h>
19
20 #define NMAX 20
21
22 void Menu(int *s);
23 void NumeroValori(int *n);
24 int CaricaVett(int *v, int n);
25 float Media(int *v, int n);
26 void CreaVett(int *v, int n, int *nuovo, int *dim, char tipo);
27 void Visualizza(int *v, int n, char tipo);
28
29 int main()
30 {
31     int scelta;           //scelta del menu'
32     int caricato = 0;     //assume il valore 1 se il vettore iniziale è stato
33                           // caricato, altrimenti assume il valore 0
34     int n;               //numero di valori da caricare nel vettore iniziale
35     int vett[NMAX];      //vettore iniziale
36     int p;               //numero di valori presenti nel vettore dei pari
37     int vettpari[NMAX];  //vettore dei numeri pari
38     int d;               //numero di valori presenti nel vettore dei dispari
39     int vettdispari[NMAX]; //vettore dei numeri dispari
40     float m;             //media
41
42     NumeroValori(&n);
43
44     do {
45         Menu(&scelta);
46
47         switch(scelta) {
48             case 1:
49                 /**
50                  Caricamento del vettore iniziale e creazione del vettore
51                  dei valori pari e del vettore dei valori dispari.
52                  **/
53                 caricato = CaricaVett(vett, n);
54                 CreaVett(vett, n, vettpari, &p, 'p');
55                 CreaVett(vett, n, vettdispari, &d, 'd');
56                 break;
57             case 2:

```

```
58     /**
59     Calcolo e visualizzazione della media del vettore iniziale.
60     **/
61     if(caricato){
62         m = Media(vett, n);
63         printf("\nLa media del vettore iniziale e' %.2f\n", m);
64     } else printf("\nCaricare il vettore iniziale\n");
65     break;
66 case 3:
67     /**
68     Visualizzazione dei valori del vettore dei pari.
69     **/
70     if(caricato)
71         Visualizza(vettpari, p, 'p');
72     else printf("\nCaricare il vettore iniziale\n");
73     break;
74 case 4:
75     /**
76     Visualizzazione dei valori del vettore dei dispari.
77     **/
78     if(caricato)
79         Visualizza(vettdispari, d, 'd');
80     else printf("\nCaricare il vettore iniziale\n");
81     break;
82 case 5:
83     /**
84     Calcolo e visualizzazione della media del vettore dei pari.
85     Viene riutilizzata la funzione Media().
86     **/
87     if(caricato){
88         m = Media(vettpari, p);
89         printf("%.2f", m);
90     } else printf("\nCaricare il vettore iniziale\n");
91     break;
92 case 6:
93     /**
94     Calcolo e visualizzazione della media del vettore dei dispari.
95     Viene riutilizzata la funzione Media().
96     **/
97     if(caricato){
98         m = Media(vettdispari, d);
99         printf("%.2f", m);
100    } else printf("\nCaricare il vettore iniziale\n");
101    break;
102 case 7:
103     break;
104 default:
105     printf("Scelta errata.\n");
106 }
107
108 }while(scelta != 7);
109
110 return 0;
111 }
112
113 /**
114 Menu()
```

```
115     Visualizza il menu'.
116     @param int *s indica il valore numerico del menu' scelto dall'utenete
117     @return la funzione non ritorna valori
118     @author mgm
119     **/
120 void Menu(int *s)
121 {
122     printf("\n** Compito del 11/04/2014 - Vettori e medie **\n\n");
123     printf("1) Caricamento vettore e creazione del vettore dei pari e dei dispari\n");
124     printf("2) Media del vettore iniziale\n");
125     printf("3) Visualizza il vettore dei pari\n");
126     printf("4) Visualizza il vettore dei dispari\n");
127     printf("5) Visualizza la media dei pari\n");
128     printf("6) Visualizza la media dei dispari\n");
129     printf("7) Esci\n\n");
130     printf("Scelta: ");
131     scanf("%d", s);
132     return;
133 }
134
135 /**
136     NumeroValori()
137     Permette di inserire il numero di valori da cariacare.
138     @param int *n numero di valori da caricare nel vettore
139     @return la funzione non ritorna valori
140     @author mgm
141     **/
142 void NumeroValori(int *n)
143 {
144     do{
145         printf("\nInserire il numero di valori da caricare: ");
146         scanf("%d", n);
147         if(*n < 1 || *n > NMAX)
148             printf("\nIl numero di valori deve essere incluso fra 1 e 20\n");
149     } while(*n < 1 || *n > NMAX);
150
151     return;
152 }
153
154 /**
155     CaricaVett()
156     Permette di inserire il numero di valori da cariacare.
157     @param int *v vettore da caricare, int n numero di valori da caricare nel
158         vettore
159     @return 1 per indicare che il vettore e' stato caricato
160     @author mgm
161     **/
162 int CaricaVett(int *v, int n)
163 {
164     int i;
165
166     printf("Inserie i valori :\n");
167     for(i = 0; i < n; i++){
168         scanf("%d", &v[i]);
169     }
170
171     return 1;
```

```
172     }
173
174     /**
175     Media()
176     Permette di calcolare la media dei valori presenti nel vettore.
177     @param int *v vettore, int n numero di valori presenti nel vettore
178     @return media dei valori presenti nel vettore
179     @author mgm
180     **/
181 float Media(int *v, int n)
182 {
183     float m;
184     int i;
185
186     for(i = 0, m = 0; i < n; i++)
187         m += v[i];
188
189     m /= n;
190
191     return m;
192 }
193
194 /**
195 CreaVett()
196 Permette di creare un nuovo vettore dai soli elementi pari o dispari del
197 vettore originale.
198 @param int *v vettore originale, int n numero di valori presenti nel
199 vettore originale, int *nuovo nuovo vettore, int *dim numero di
200 valori presenti nel nuovo vettore, char tipo specifica se si vuole
201 creare il vettore dei valori pari o quello dei valori dispari
202 @return la funzione non ritorna valori
203 @author mgm
204 **/
205 void CreaVett(int *v, int n, int *nuovo, int *dim, char tipo)
206 {
207     int i, j;
208
209     for(i = 0, j = 0; i < n; i++){
210         /**
211          La condizione dell'if() permettera' di creare un solo tipo di vettore,
212          a seconda del valore contenuto nella variabile tipo.
213          **/
214         if((!(v[i] % 2) && (tipo == 'p')) || ((v[i] % 2) && (tipo == 'd'))){
215             nuovo[j] = v[i];
216             j++;
217         }
218     }
219
220     *dim = j;
221
222     return;
223 }
224
225 /**
226 Visualizza()
227 Visualizza i valori presenti nel vettore.
228 @param int *v vettore, int n numero di valori presenti nel vettore, char
```

```
229         tipo specifica se si vuole creare il vettore dei valori pari o quello
230         dei valori dispari
231     @return media dei valori presenti nel vettore
232     @author mgm
233     **/
234 void Visualizza(int *v, int n, char tipo)
235 {
236     int i;
237
238     printf("\nElementi ");
239     if(tipo == 'p')
240         printf("pari\n");
241     else
242         printf("dispari\n");
243
244     for(i = 0; i < n; i++)
245         printf("%d ", *(v + i));
246
247     return;
248 }
249
```