static void caricaVet(int[] v, int n) //caricamento vettore

{

for (int i = 0; i < n; i++)

v[i] = rnd.Next(1, 11);

}

static void stampaVet(int[] v, int n, string nome)

{

Console.WriteLine(nome);

for (int i = 0; i < n; i++)

Console.WriteLine("Posizione " + i + " :" + v[i]);

}

static void prodottoVet() //prodotto vettoriale

{

int[] a;

int[] b;

int[] c;

int n;

do

{

Console.Write("Inserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

a = new int[n];

b = new int[n];

c = new int[n];

caricaVet(a, n);

caricaVet(b, n);

for (int i = 0; i < n; i++)

c[i] = a[i] \* b[i];

stampaVet(a, n, "\nVettore A\n");

stampaVet(b, n, "\nVettore B\n");

stampaVet(c, n, "\nVettore C\n");

Console.ReadKey(true);

}

static void copiaVet() //copia vettore a in vettore b

{

int[] a;

int[] b;

int n;

do

{

Console.Write("\n\nInserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

a = new int[n];

b = new int[n];

caricaVet(a, n);

for (int i = 0; i < n; i++)

b[i] = a[i];

Console.WriteLine("\n\nstampa vettore A\n\n");

stampaVet(a, n);

Console.WriteLine("\n\nstampa vettore A\n\n");

stampaVet(b, n);

Console.ReadKey(true);

}

static void copiaVetInverso() //copia vettore a inverso in vettore b

{

int[] a;

int[] b;

int n;

do

{

Console.Write("\n\nInserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

int j = n - 1;

a = new int[n];

b = new int[n];

caricaVet(a, n);

for (int i = 0; i < n; i++)

b[j--] = a[i];

stampaVet(a, n, "\nVettore A\n");

stampaVet(b, n, "\nVettore B\n");

Console.ReadKey(true);

}

static void cercaMAX() //cerca numero più grande

{

int[] a;

int n,m=0;

do

{

Console.Write("Inserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

a = new int[n];

caricaVet(a, n);

stampaVet(a, n);

for (int i = 0; i < n; i++)

if (a[i]>m)

m = a[i];

Console.WriteLine("Il valore più grande è: " + m);

Console.ReadKey(true);

}

static void invertiVet() //inverti vettore su se stesso

{

int[] a;

int b;

int n;

do

{

Console.Write("\n\nInserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

int j = n - 1;

a = new int[n];

caricaVet(a, n);

stampaVet(a, n, "\nVettore A\n");

for (int i = 0; i < n/2; i++)

{

b = a[j];

a[j--] = a[i];

a[i] = b;

}

stampaVet(a, n, "\nVettore A\n");

Console.ReadKey(true);

}

static void prodottoScalareVet() //prodotto scalare vettori

{

int[] a;

int[] b;

int n,ps=0;

do

{

Console.Write("\n\nInserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

a = new int[n];

b = new int[n];

caricaVet(a, n);

caricaVet(b, n);

for (int i = 0; i < n; i++)

{

ps = a[i] \* b[i] + ps;

}

stampaVet(a, n, "\nVettore A\n");

stampaVet(b, n, "\nVettore B\n");

Console.WriteLine("Prodotto scalare= " + ps);

Console.ReadKey(true);

}

private static void controlloOrdinamentoVet()

{

int[] a;

int m = int.MinValue;

int n;

do

{

Console.Write("Inserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

a = new int[n];

caricaVet(a, n);

stampaVet(a, n, "\nVettore A\n");

for (int i = 0; i < n; i++)

{

if (a[i] > m)

{

m = a[i];

}

else

{

i = n;

Console.WriteLine("Vettore non ordinato");

m = -1;

}

}

if (m != -1)

Console.WriteLine("Vettore ordinato");

Console.ReadKey();

}

private static void contaOccorenzeVet()

{

int[] a;

int m=0,x;

int n;

do

{

Console.Write("Inserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

do

{

Console.Write("Inserire numero da cercare: ");

if (!int.TryParse(Console.ReadLine(), out x))

Console.WriteLine("Input errato");

} while (x <= 0 || x>101);

a = new int[n];

caricaVet(a, n);

stampaVet(a, n, "\nVettore A\n");

for (int i = 0; i < n; i++)

if (a[i] == x)

m++;

Console.WriteLine("Il numero " + x + " compare " + m + " volte");

Console.ReadKey();

}

private static void sommaDispariPari()

{

int[] a;

int p = 0, d=0;

int n;

do

{

Console.Write("Inserire lunghezza vettore: ");

if (!int.TryParse(Console.ReadLine(), out n))

Console.WriteLine("Input errato");

} while (n <= 0);

a = new int[n];

caricaVet(a, n);

stampaVet(a, n, "\nVettore A\n");

for (int i = 0; i < n; i++)

if (a[i]%2==1)

d += a[i];

else

p += a[i];

Console.WriteLine("La somma dei numeri pari è: "+p);

Console.WriteLine("La somma dei numeri dispari è: " + d);

Console.ReadKey();

}