

LABORATORIO DI PROGRAMMAZIONE E CALCOLO

Canale 1 (A-K) A.A. 2024/2025

Foglio di esercizi n. 7

1. Scrivere un programma C++ strutturato in funzioni che acquisisce da tastiera un numero intero positivo $n \leq 5$, gli elementi di una matrice $\mathbf{A} = (a_{i,j}) \in \mathbb{R}^{n \times n}$ - che dovranno essere introdotti per righe (prima gli elementi della prima riga, poi quelli della seconda, ecc.) - e le componenti di un vettore $\mathbf{u} = (u_j) \in \mathbb{R}^n$; successivamente calcola e stampa sul video il vettore $\mathbf{v} = (v_j) \in \mathbb{R}^n$ tale che $\mathbf{v} = \mathbf{A}\mathbf{u}$, il vettore $\mathbf{w} = (w_j) \in \mathbb{R}^n$ contenente elemento per elemento la somma delle componenti della corrispondente colonna di \mathbf{A} , ovvero $w_j = \sum_{i=1}^n a_{i,j}$.

```
#include <stdio.h>
#define N 5
int lettura_mat(float[][N]);
void lettura_vet(float[],int);
void stampa_vet(float v[],int);
void somma_colonne(float[][N],float[],int);
void prod_mat_vet(float[][N],float[],float[],int);

int main()
{
    float a[N][N], u[N], v[N]={0}, w[N]={0};
    int n = lettura_mat(a);
    printf("\nInserire le componenti del vettore u.\n");
    lettura_vet(u,n);
    prod_mat_vet(a,u,v,n);
    printf("\nProdotto matrice*vettore: \n");
    stampa_vet(v,n);
    printf("\nVettore somma colonne della matrice: \n");
    somma_colonne(a,w,n);
    stampa_vet(w,n);
    return 0;
}

void lettura_vet(float u[],int d)
{
    for (int i=0;i<d;i++)
    {
        printf("u[%d] = ",i+1);
        scanf("%f",&u[i]);
    }
    return;
}
```

```

int lettura_mat(float mat[][N])
{
    int n;
    printf("\nInserire la dimensione n<=%d:\n",N);
    do {printf("n = ");scanf("%d",&n);}
    while (n>N||n<=0);
    printf("\nInserire le componenti della matrice.\n");
    for (int i=0;i<n;i++)
        for (int j=0;j<n;j++)
        {
            printf("A[%d,%d] = ",i+1,j+1);
            scanf("%f",&mat[i][j]);
        }
    return n;
}

void stampa_vet(float v[],int d)
{
    printf("[ ");
    for (int i=0;i<d;i++) printf("%f ",v[i]);
    printf("]^T\n");
    return;
}

void somma_colonne(float mat[][N],float v[],int d)
{
    for (int j=0; j<d; j++)
        for (int i=0; i<d; i++) v[j]+=mat[i][j];
    return;
}

void prod_mat_vet(float a[][N],float u[],float v[],int d)
{
    for (int i=0;i<d;i++)
        for (int j=0;j<d;j++) v[i]+=a[i][j]*u[j];
    return;
}

```

2. Scrivere un programma C++ strutturato in funzioni che costruisce la seguente matrice:

$$\mathbf{A} = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

e ne calcola e stampa sul video la somma s e il prodotto p degli elementi diagonali, e i valori n_1 e n_2 definiti come segue:

$$n_1 = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{i,j}|, \quad n_2 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{i,j}|.$$

3. Scrivere un programma C++ strutturato in funzioni che calcola il prodotto righe per colonne $\mathbf{C} = \mathbf{AB}$, dove \mathbf{A} è la matrice definita nell'esercizio precedente e \mathbf{B} è la seguente matrice:

$$\mathbf{B} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

e stampa sul video la matrice \mathbf{C} .

4. Scrivere un programma C++ strutturato in funzioni che acquisisce da tastiera un numero intero positivo $n \leq 50$, costruisce una matrice triangolare inferiore \mathbf{L} di n righe ed n colonne ad elementi interi random tra 1 e 15 e un vettore \mathbf{b} di dimensione n contenente elemento per elemento la somma delle componenti della corrispondente riga di \mathbf{L} ; successivamente risolve il sistema lineare $\mathbf{Lx}=\mathbf{b}$ e stampa sul video la somma delle componenti del vettore \mathbf{x} (che dovrebbe essere uguale a n , dato che le componenti di \mathbf{x} dovrebbero essere tutte uguali a 1).