

Esercizio 1

```

#include <stdio.h>
#include <stdlib.h>
#define V 2

int main(int argc, char *argv[])
{
    int i,vmax=18, vmin=30, voto, somma=0;
    for (i=1; i<=V; i++){
        do{
            printf("Inserire il voto n. %d: ", i);
            scanf("%d",&voto);

            }while (voto>31 || voto<18);
            if (voto<vmin) vmin=voto;
            if (voto>vmax) vmax=voto;
            somma+=voto;
        }

        printf("Media = %f\n", (float) somma/V);
        printf("V. Max = %d\n", vmax);
        printf("V. Min = %d\n", vmin);
        system("PAUSE");
        return 0;
    }
}

```

Esercizio 2

```

#include <stdio.h>
main() {
    int x, y, t;

    do {
        printf("Inserire due interi maggiori di zero:");
        scanf("%d %d", &x, &y);
    } while (x <= 0 OR || y <= 0);

    while (x != y) {
        if (x < y) {
            t = x;
            x = y;
            y = t;
        }
        x = x - y;
    }

    printf("Risultato ottenuto = %d \n", &x);
}

```

Soluzione: il programma calcola il massimo comun divisore di due numeri (applicando l'algoritmo di Euclide)

Esercizio 3

```
int sommaNumeriPari(int n){
    int i = 2,somma=0;
    while (i<=n) {
        somma+=i; i=i+2;
    }
    return somma;
}
```

```
int sommaNumeriPari(int n){
    sommaNumeriPari_TAIL(n,2,0)
}
```

```
int sommaNumeriPari_TAIL(int n, int i, int somma){
    if (i<=n) {
        somma+=i; i=i+2;
        return sommaNumeriPari_TAIL(n, i, somma)
    }
    return somma;
}
```

Esercizio 4

x = y+3;

l'espressione cambia in 8 il valore di x

- il simbolo y **a destra** dell'operatore = denota **il valore attuale (R-value) di y**, cioè 5
- il simbolo x **a sinistra** dell'operatore = denota **la cella di memoria associata a x (L-value)**, a cui viene assegnato il valore dell'espressione di destra (8)

l'**espressione** nel suo complesso denota il **valore della variabile** dopo la modifica, cioè 8