

```

prompt= ('introduce numero de iteraciones: ');
n= input(prompt);
prompt= ('introduce el valor inicial de x0: ');
a= input (prompt);
vectorIt= 1:1:n

```

```

vectorIt = 1×100
    1     2     3     4     5     6     7     8     9    10    11    12    13 ...

```

```

for i=1:1:n
    if i==1
        g(i)= cos(a)/3
    else
        g(i)= cos(g(i-1))/3
        if i==20
            if abs(g(i-1)-g(i))>0.01
                break;
            end
        end
    end
end
end

```

```

g = -0.2179
g = 1×2
    -0.2179     0.3255
g = 1×3
    -0.2179     0.3255     0.3158
g = 1×4
    -0.2179     0.3255     0.3158     0.3168
g = 1×5
    -0.2179     0.3255     0.3158     0.3168     0.3167
g = 1×6
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168
g = 1×7
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168
g = 1×8
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168
g = 1×9
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×10
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×11
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×12
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×13
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×14
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×15
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×16
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×17
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×18
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...
g = 1×19
    -0.2179     0.3255     0.3158     0.3168     0.3167     0.3168     0.3168     0.3168 ...

```



```

g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...
g =
-0.2179    0.3255    0.3158    0.3168    0.3167    0.3168    0.3168    0.3168 ...

```

```

if size(g)==size(vectorIt)
    disp('el algoritmo termino satisfactoriamente');
else
    disp('el algoritmo no converge');
end

```

```

el algoritmo termino satisfactoriamente

```