%Q1

clear all, clc

len=input('donner taille du vecteur')

for i= 1:len;

vect(i)=input('donner une valeur')

end %or by vectorization randint([10 20],4,5)

av=0;

for i=1:len ;

av=av+vect(i);

end

av=av/len;

StDv=0; %The standard deviation is the square root of the variance.

for i=1:len;

StDv= StDv+(vect(i)-av)^2;

end

StDv=StDv/((len)-1);

StDv=sqrt(StDv);

%display results(or see on the workspace )

av

StDv

%compare by vectorization

mean(vect)

std(vect)

function[StDv,av]=quest1(vect)

len=length(vect)

for i= 1:len;

vect(i)=input('donner une valeur')

end %or by vectorization randint([10 20],4,5)

av=0;

for i=1:len ;

av=av+vect(i);

end

av=av/len;

StDv=0; %The standard deviation is the square root of the variance.

for i=1:len;

StDv= StDv+(vect(i)-av)^2;

end

StDv=StDv/((len)-1);

StDv=sqrt(StDv);

End

%Q2

N=input('type an integer')

if fix(N)==N

S=0 % N==0

if (N<0)

for i=N:-1

S=S+i

end

else if (N>0)

for i=1:N

S=S+i

end

S=S+N^2

end

end

else fprintf('%f is not an integer',N)

end

function[S]=test(N)

if fix(N)==N

S=0 % N==0

if (N<0)

for i=N:-1

S=S+i

end

else if (N>0)

for i=1:N

S=S+i

end

S=S+N^2

end

end

else fprintf('%f is not an integer',N)

end

end

%Q3

V=input('type a vector') %[2 4 7]

if (length(V)==numel(V) & length(V)>=2) % it is a vector an there is enough

% elements to compare

i=1;

while ((i<=length(V)-1) & (V(i)<=V(i+1)))

i=i+1;

end

if i==length(V);

fprintf('it is sorted in ascending order')

else

fprintf('the vector is not sorted in ascending order since %d',i+1)

end

else fprintf('it is not a vector of 2 elements at least')

end

function[]=quest3(V)

if (length(V)==numel(V) & length(V)>=2) % it is a vector an there is enough

% elements to compare

i=1;

while ((i<=length(V)-1) & (V(i)<=V(i+1)))

i=i+1;

end

if i==length(V);

fprintf('it is sorted in ascending order')

else

fprintf('the vector is not sorted in ascending order since %d',i+1)

end

else fprintf('it is not a vector of 2 elements at least')

end

end

%Q4

V=input('type a vector') %[2 4 7]

VPOS=[]; %to create an empty vector

VNEG=[]; %and it is required to use 'end' at the first time

if (length(V)==numel(V) & length(V)~=0) %it is a vector?

for i=1:length(V)

if V(i)>=0

VPOS(end+1)=V(i);

else

VNEG(end+1)=V(i);

end

end

VPOS

VNEG

else fprintf('it is not a vector')

end

function[VPOS,VNEG]=question4(V)

VPOS=[]; %to create an empty vector

VNEG=[]; %and it is required to use 'end' at the first time

if (length(V)==numel(V) & length(V)~=0) %it is a vector?

for i=1:length(V)

if V(i)>=0

VPOS(end+1)=V(i);

else

VNEG(end+1)=V(i);

end

end

VPOS

VNEG

else fprintf('it is not a vector')

end

end

%Q5

M=input('type a matrix')

for i=1:size(M,1)-1

Cpt(i)=0

for j=1:size(M,2)

k=1

while (k<=size(M,2)& (~isequal(M(i,j),M(i+1,k))) )

k=k+1

end

if k==size(M,2)+1

Cpt(i) = Cpt(i)+1

end

end

end

function[cpt=quest5(M)

for i=1:size(M,1)-1

Cpt(i)=0

for j=1:size(M,2)

k=1

while (k<=size(M,2)& (~isequal(M(i,j),M(i+1,k))) )

k=k+1

end

if k==size(M,2)+1

Cpt(i) = Cpt(i)+1

end

end

end

end

%Q6

for x=0:10

for y=0:20

f(x+1,y+1)=x^2\*sin(y)

end

end

function[f]=quest6(x,y)

for x=0:10

for y=0:20

f(x+1,y+1)=x^2\*sin(y)

end

end

end