/\*

Name: Prisha Patel

Program Number: COP 3223

Title: Stats.c

Date: 3/20/2023

\*/

#include <stdio.h>

#include <math.h>

void readGrades(double grades[], int \*n);

void frequency(double grades[], int n);

int maximum(double grades[], int n);

int minimum(double grades[], int n);

void deleteElement(double grades[], int \*n, int loc);

double mean(double grades[], int n);

double StandardDeviation(double grades[], int n);

// Function to take user input of grades and create an array

void readGrades(double grades[], int \*n)

{

    int x = 0;

    while(grades [x - 1] != -999){

        scanf("%lf", &grades[x]);

        x++;

    }

    \*n= x - 1;

}

// Function to create the histogram graph

void frequency(double grades[], int n)

{

    int interval [100] = {0};

    printf("\nHere is the histogram of the adjusted data:\n\n");

    for (int x = 0; x < n; x++)

    {

        if (grades[x] >= 0 && grades[x] <= 4)

        {

            interval[0]++;

        }

        else if (grades[x] >= 5 && grades[x] <= 9)

        {

            interval[1]++;

        }

        else if (grades[x] >= 10 && grades[x] <= 14)

        {

            interval[2]++;

        }

        else if (grades[x] >= 15 && grades[x] <= 19)

        {

            interval[3]++;

        }

        else if (grades[x] >= 20 && grades[x] <= 24)

        {

            interval[4]++;

        }

        else if (grades[x] >= 25 && grades[x] <= 29)

        {

            interval[5]++;

        }

        else if (grades[x] >= 30 && grades[x] <= 34)

        {

            interval[6]++;

        }

        else if (grades[x] >= 35 && grades[x] <= 39)

        {

            interval[7]++;

        }

        else if (grades[x] >= 40 && grades[x] <= 44)

        {

            interval[8]++;

        }

        else if (grades[x] >= 45 && grades[x] <= 49)

        {

            interval[9]++;

        }

        else if (grades[x] >= 50 && grades[x] <= 54)

        {

            interval[10]++;

        }

        else if (grades[x] >= 55 && grades[x] <= 59)

        {

            interval[11]++;

        }

        else if (grades[x] >= 60 && grades[x] <= 64)

        {

            interval[12]++;

        }

        else if (grades[x] >= 65 && grades[x] <= 69)

        {

            interval[13]++;

        }

        else if (grades[x] >= 70 && grades[x] <= 74)

        {

            interval[14]++;

        }

        else if (grades[x] >= 75 && grades[x] <= 79)

        {

            interval[15]++;

        }

        else if (grades[x] >= 80 && grades[x] <= 84)

        {

            interval[16]++;

        }

        else if (grades[x] >= 85 && grades[x] <= 89)

        {

            interval[17]++;

        }

        else if (grades[x] >= 90 && grades[x] <= 94)

        {

            interval[18]++;

        }

        else if (grades[x] >= 95 && grades[x] <= 99)

        {

            interval[19]++;

        }

        else if (grades[x] == 100)

        {

            interval[20]++;

        }

    }

    // print

    printf("0 - 4|");

    for (int x = 0; x < interval[0]; x++)

    {

        printf("\*");

    }

    printf("\n5 - 9|");

    for (int x = 0; x < interval[1]; x++)

    {

        printf("\*");

    }

    printf("\n10-14|");

    for (int x = 0; x < interval[2]; x++)

    {

        printf("\*");

    }

    printf("\n15-19|");

    for (int x = 0; x < interval[2]; x++)

    {

        printf("\*");

    }

    printf("\n20-24|");

    for (int x = 0; x < interval[4]; x++)

    {

        printf("\*");

    }

    printf("\n25-29|");

    for (int x = 0; x < interval[5]; x++)

    {

        printf("\*");

    }

    printf("\n30-34|");

    for (int x = 0; x < interval[6]; x++)

    {

        printf("\*");

    }

    printf("\n35-39|");

    for (int x = 0; x < interval[7]; x++)

    {

        printf("\*");

    }

    printf("\n40-45|");

    for (int x = 0; x < interval[8]; x++)

    {

        printf("\*");

    }

    printf("\n46-49|");

    for (int x = 0; x < interval[9]; x++)

    {

        printf("\*");

    }

    printf("\n50-54|");

    for (int x = 0; x < interval[10]; x++)

    {

        printf("\*");

    }

    printf("\n55-59|");

    for (int x = 0; x < interval[11]; x++)

    {

        printf("\*");

    }

    printf("\n60-64|");

    for (int x = 0; x < interval[12]; x++)

    {

        printf("\*");

    }

    printf("\n65-69|");

    for (int x = 0; x < interval[13]; x++)

    {

        printf("\*");

    }

    printf("\n70-74|");

    for (int x = 0; x < interval[14]; x++)

    {

        printf("\*");

    }

    printf("\n75-79|");

    for (int x = 0; x < interval[15]; x++)

    {

        printf("\*");

    }

    printf("\n80-84|");

    for (int x = 0; x < interval[16]; x++)

    {

        printf("\*");

    }

    printf("\n85-89|");

    for (int x = 0; x < interval[17]; x++)

    {

        printf("\*");

    }

    printf("\n90-94|");

    for (int x = 0; x < interval[18]; x++)

    {

        printf("\*");

    }

    printf("\n95-99|");

    for (int x = 0; x < interval[19]; x++)

    {

        printf("\*");

    }

    printf("\n  100|");

    for (int x = 0; x < interval[20]; x++)

    {

        printf("\*");

    }

    printf("\n");

}

// Function to return the maximum number in the array

int maximum(double grades[], int n)

{

    int max = 0;

    for (int x = 1; x < n ; x++)

    {

        if(grades[x] > grades[max])

        {

            max = x;

        }

    }

    return max;

}

// Function to return the minimun number in the array

int minimum(double grades[], int n)

{

    int min = 0;

    for (int x = 1; x < n ; x++)

    {

        if(grades[x] < grades[min])

        {

            min = x;

        }

    }

    return min;

}

// Function to go to the location of the element needed to be deleted

void deleteElement(double grades[], int \*n, int loc)

{

    for (int x = loc; x < \*n - 1; x++)

    {

        grades[x] = grades[x + 1];

    }

    \*n -= 1;

}

// Function to calculate the Mean

double mean(double grades[], int n)

{

    double sum = 0;

    for (int x = 0; x < n; x++)

    {

        sum += grades[x];

    }

    return sum / n;

}

// Function to calculate the Standard Deviation

double StandardDeviation(double grades[], int n)

{

    double meanGrade = mean(grades,n);

    double sum =0;

    for (int x = 0; x < n; x++)

    {

        sum += pow(grades[x] - meanGrade, 2);

    }

    return sqrt(sum / n);

}

// Main Function

int main()

{

    double grades[100] = {0};

    int n;

    readGrades(grades, &n);

    int min;

    min = minimum(grades, n);

    printf("\nThe data has been adjusted by removing the minimun: %.2lf \n", grades[min]);

    deleteElement(grades, &n, min);

    int max;

    max = maximum(grades,n);

    printf("The data has been adjusted by removing the maximum: %.2lf \n", grades[max]);

    deleteElement(grades, &n, max);

    double meanGrade = mean(grades,n);

    printf("The adjusted mean is: %.2lf \n", meanGrade);

    double standardDev = StandardDeviation(grades,n);

    printf("The adjusted standard deviation is: %.2lf\n", standardDev);

    frequency(grades,n);

    return 0;

}

Text

Description automatically generated