**SSN COLLEGE OF ENGINEERING (Autonomous)**

**Affiliated to Anna University**

**DEPARTMENT OF CSE**

**UCS 1211 PROGRAMMING IN C LABORATORY**

**A3: Array handling in C**

========================================================

**REGISTRATION NO: 185001112**

**NAME: PRATHYUSH S**

**CLASS: CSE-B (SEMESTER-2)**

**========================================================**

**1. The number 138 is called well ordered because the digits in the number (1,3,8) increase from left to right. (1 < 3 < 8). The number 365 is not well ordered. Write a progRam that will find and display all possible three digit well-ordered numbers. Also display the total number of three digit well-ordered numbers. Make use of Arrays.**

#include<stdio.h>

void split(int a[3],int i);

int order(int a[3]);

void main()

{

int i,j,b,a[3],count=0;

printf("Ordered numbers are:\n");

for(i=100;i<1000;i++)

{

split(a,i);

b=order(a);

if(b==1)

{

printf("%d\t",i);

count++;

}

}

printf("\nTotal number of three digit ordered numbers : %d\n",count);

}

void split(int a[3],int i)

{

int r,j,k=2;

while(i!=0)

{

r=i%10;

for(j=k;j>=0;j--)

{

a[j]=r;

break;

}

k--;

i=i/10;

}

}

int order(int a[3])

{

if((a[0]<a[1]) && (a[1]<a[2]))

return 1;

else

return 0;

}

Output:

cseb112@jtl-10:gcc ass31.c –o ass31

cseb112@jtl-10:./ass31

Ordered numbers are:

123 124 125 126 127 128 129 134 135 136 137 138 139 145 146 147 148 149 156 157 158 159 167 168 169 178 179 189 234 235 236 237 238 239 245 246 247 248 249 256 257 258 259 267 268 269 278 279 289 345 346 347 348 349 356 357 358 359 367 368 369 378 379 389 456 457 458 459 467 468 469 478 479 489 567 568 569 578 579 589 678 679 689 789

Total number of three digit ordered numbers : 84

**2. Write a progRam that accepts a set of digits (0 to 9) as input and prints a vertical histogRam representing the occurrences of each digit. Test your progRam with the set of 13 digits: 1, 7, 2 , 9, 6, 7, 1, 3, 7, 5, 7, 9, 0**

#include<stdio.h>

void main()

{

int i,n,a[10],b,j,max=0,k;

printf("Enter the number of terms :\n");

scanf("%d",&n);

for(i=0;i<10;i++)

{

a[i]=0;

}

printf("Enter the numbers:\n");

for(i=0;i<n;i++)

{

scanf("%d",&b);

if(b>=0 && b<=9)

{

a[b]++;

if(a[b]>max)

max=a[b];

}

else

printf("Enter a number between 0 and 9\n");

}

k=max;

printf("Vertical HistogRam:\n");

for(i=1;i<=max;i++)

{

for(j=0;j<10;j++)

{

if(a[j]>=k)

printf("\*");

else

printf(" ");

}

printf("\n");

k--;

}

for(i=0;i<10;i++)

{

printf("%d",i);

}

}

Output:

cseb112@jtl-10:gcc ass32.c –o ass32

cseb112@jtl-10:./ass32

Enter the number of terms :

12

Enter the numbers:

1

7

2

9

6

7

1

3

7

5

7

9

Vertical HistogRam:

\*

\*

\* \* \*

\*\*\* \*\*\* \*

0123456789

**3. Given an array of integers, push all the zero’s of a given array to the end of the array. Write a progRam in C that implements the function pushZerosToEnd(int arr[], int n).**

#include<stdio.h>

void push(int b[], int a);

void main()

{

int a,i;

printf("Enter the number of elements :\n");

scanf("%d",&a);

int b[a];

printf("Enter the elements:\n");

for(i=0;i<a;i++)

{

scanf("%d",&b[i]);

}

printf("Input :");

for(i=0;i<a;i++)

{

printf("%d ",b[i]);

}

push(b,a);

printf("\nOutput :");

for(i=0;i<a;i++)

{

printf("%d ",b[i]);

}

}

void push(int b[], int a)

{

int i,k=0;

for(i=0;i<a;i++)

{

if(b[i]!=0)

{

b[k]=b[i];

k++;

}

}

for(i=k;i<a;i++)

{

b[i]=0;

}

}

Output:

cseb112@jtl-10:gcc ass33.c –o ass33

cseb112@jtl-10:./ass33

Enter the number of elements :

8

Enter the elements:

5

3

0

1

3

0

8

0

Input :5 3 0 1 3 0 8 0

Output :5 3 1 3 8 0 0 0

**4. Write an interactive C progRam to process the exam scores for a group of students in a C progRamming course. Begin by specifying the number of exam scores for each student (assume this value is the same for all students in the class). Then enter each student’s name and exam scores. Calculate an average score for each student, and an overall class average (an average of the individual student averages). Display the overall class average, followed by the name, the individual exam scores and the average score for each student. Store the student names in a twodimensional character array, and store the exam scores in a two-dimensional floating-point array. Make the progRam as general as possible. Label the output clearly.**

#include<stdio.h>

void main()

{

int n,m,i,j,k;

printf("Enter the number of students:\n");

scanf("%d",&n);

char name[n][100];

printf("Enter the names:\n");

for(i=0;i<n;i++)

{

scanf("%s",&name[i]);

}

printf("Enter the number of exams:");

scanf("%d",&m);

float mark[n][m];

for(i=0;i<n;i++)

{

for(j=0;j<m;j++)

{

printf("Enter %s's %d exam marks : \n",name[i],j+1);

scanf("%f",&mark[i][j]);

}

}

float avg[n][1],cls=0,clsavg=0;

for(i=0;i<n;i++)

{

float sum=0;

for(j=0;j<m;j++)

sum+=mark[i][j];

avg[i][0]=(sum/m);

}

for(i=0;i<n;i++)

{

cls+=avg[i][0];

}

clsavg=cls/n;

printf("Name Marks\n");

printf(" ");

for(i=1;i<m+1;i++)

printf("%d ",i);

printf("Average");

printf("\n");

k=0;

for(i=0;i<n;i++)

{

printf("%-13s",name[i]);

for(j=0;j<m;j++)

{

printf("%5.0f ",mark[i][j]);

}

for(j=k;j<k+1;j++)

{

printf(" %.2f ",avg[j][0]);

}

k++;

printf("\n");

}

printf("Class Average = %.2f\n",clsavg);

}

Output:

cseb112@jtl-10:gcc ass34.c –o ass34

cseb112@jtl-10:./ass34

Enter the number of students:

5

Enter the names:

Praveen

Prakash

Hariharan

Dhaneesh

Ram

Enter the number of exams:6

Enter Praveen's 1 exam marks :

45

Enter Praveen's 2 exam marks :

80

Enter Praveen's 3 exam marks :

80

Enter Praveen's 4 exam marks :

95

Enter Praveen's 5 exam marks :

55

Enter Praveen's 6 exam marks :

75

Enter Prakash's 1 exam marks :

60

Enter Prakash's 2 exam marks :

50

Enter Prakash's 3 exam marks :

70

Enter Prakash's 4 exam marks :

75

Enter Prakash's 5 exam marks :

55

Enter Prakash's 6 exam marks :

80

Enter Hariharan's 1 exam marks :

40

Enter Hariharan's 2 exam marks :

30

Enter Hariharan's 3 exam marks :

10

Enter Hariharan's 4 exam marks :

45

Enter Hariharan's 5 exam marks :

60

Enter Hariharan's 6 exam marks :

55

Enter Dhaneesh's 1 exam marks :

0

Enter Dhaneesh's 2 exam marks :

5

Enter Dhaneesh's 3 exam marks :

5

Enter Dhaneesh's 4 exam marks :

0

Enter Dhaneesh's 5 exam marks :

10

Enter Dhaneesh's 6 exam marks :

5

Enter Ram's 1 exam marks :

95

Enter Ram's 2 exam marks :

90

Enter Ram's 3 exam marks :

80

Enter Ram's 4 exam marks :

95

Enter Ram's 5 exam marks :

85

Enter Ram's 6 exam marks :

80

Name Marks

1 2 3 4 5 6 Average

Praveen 45 80 80 95 55 75 71.67

Prakash 60 50 70 75 55 80 65.00

Hariharan 40 30 10 45 60 55 40.00

Dhaneesh 0 5 5 0 10 5 4.17

Ram 95 90 80 95 85 80 87.50

Class Average = 53.67

**a) Modify this progRam to allow for unequal weighting of the individual exam scores. In particular, assume that each of the first four exams contributes 15 percent to the final score, and each of the last two exams contributes 20 percent.**

#include<stdio.h>

void main()

{

int n,m,i,j,k;

printf("Enter the number of students:\n");

scanf("%d",&n);

char name[n][100];

printf("Enter the names:\n");

for(i=0;i<n;i++)

{

scanf("%s",&name[i]);

}

printf("Enter the number of exams:");

scanf("%d",&m);

float mark[n][m];

for(i=0;i<n;i++)

{

for(j=0;j<m;j++)

{

printf("Enter %s's %d exam marks : \n",name[i],j+1);

scanf("%f",&mark[i][j]);

}

}

float avg[n][1],cls=0,clsavg=0;

for(i=0;i<n;i++)

{

float sum=0;

for(j=0;j<m;j++)

if(j<4)

sum+=0.15\*mark[i][j];

else

sum+=0.20\*mark[i][j];

avg[i][0]=(sum);

}

for(i=0;i<n;i++)

{

cls+=avg[i][0];

}

clsavg=cls/n;

printf("Name Marks\n");

printf(" ");

for(i=1;i<m+1;i++)

printf("%d ",i);

printf("Average");

printf("\n");

k=0;

for(i=0;i<n;i++)

{

printf("%-13s",name[i]);

for(j=0;j<m;j++)

{

printf("%5.0f ",mark[i][j]);

}

for(j=k;j<k+1;j++)

{

printf(" %.2f ",avg[j][0]);

}

k++;

printf("\n");

}

printf("Class Average = %.2f\n",clsavg);

}

Output:

cseb112@jtl-10:gcc ass341.c –o ass341

cseb112@jtl-10:./ass341

Enter the number of students:

3

Enter the names:

Praveen

Prakash

Hariharan

Enter the number of exams:6

Enter Praveen's 1 exam marks :

45

Enter Praveen's 2 exam marks :

80

Enter Praveen's 3 exam marks :

80

Enter Praveen's 4 exam marks :

95

Enter Praveen's 5 exam marks :

55

Enter Praveen's 6 exam marks :

75

Enter Prakash's 1 exam marks :

60

Enter Prakash's 2 exam marks :

50

Enter Prakash's 3 exam marks :

70

Enter Prakash's 4 exam marks :

75

Enter Prakash's 5 exam marks :

55

Enter Prakash's 6 exam marks :

80

Enter Hariharan's 1 exam marks :

40

Enter Hariharan's 2 exam marks :

30

Enter Hariharan's 3 exam marks :

10

Enter Hariharan's 4 exam marks :

45

Enter Hariharan's 5 exam marks :

60

Enter Hariharan's 6 exam marks :

55

Name Marks

1 2 3 4 5 6 Average

Praveen 45 80 80 95 55 75 71.00

Prakash 60 50 70 75 55 80 65.25

Hariharan 40 30 10 45 60 55 41.75

Class Average = 59.33

**b) Extend the progRam so that the deviation of each student's average from the overall class average can be determined. Display the class average, followed by each student's name, individual exam scores, final score, and the deviation from the class average. Be sure that the output is logically organized and clearly labeled.**

#include<stdio.h>

void main()

{

int n,m,i,j,k;

printf("Enter the number of students:\n");

scanf("%d",&n);

char name[n][100];

printf("Enter the names:\n");

for(i=0;i<n;i++)

{

scanf("%s",&name[i]);

}

printf("Enter the number of exams:");

scanf("%d",&m);

float mark[n][m];

for(i=0;i<n;i++)

{

for(j=0;j<m;j++)

{

printf("Enter %s's %d exam marks : \n",name[i],j+1);

scanf("%f",&mark[i][j]);

}

}

float avg[n][1],cls=0,clsavg=0,dev[n][1];

for(i=0;i<n;i++)

{

float sum=0;

for(j=0;j<m;j++)

if(j<4)

sum+=0.15\*mark[i][j];

else

sum+=0.20\*mark[i][j];

avg[i][0]=(sum);

}

for(i=0;i<n;i++)

{

cls+=avg[i][0];

}

clsavg=cls/n;

for(i=0;i<n;i++)

{

dev[i][0]=avg[i][0]-clsavg;

}

printf("Name Marks\n");

printf(" ");

for(i=1;i<m+1;i++)

printf("%d ",i);

printf("Average");

printf(" Deviation");

printf("\n");

k=0;

for(i=0;i<n;i++)

{

printf("%-13s",name[i]);

for(j=0;j<m;j++)

{

printf("%5.0f ",mark[i][j]);

}

for(j=k;j<k+1;j++)

{

printf(" %.2f ",avg[j][0]);

printf(" %.2f ",dev[j][0]);

}

k++;

printf("\n");

}

printf("Class Average = %.2f\n",clsavg);

}

Output:

cseb112@jtl-10:gcc ass342.c –o ass342

cseb112@jtl-10:./ass342

Enter the number of students:

3

Enter the names:

Praveen

Prakash

Hariharan

Enter the number of exams:6

Enter Praveen's 1 exam marks :

45

Enter Praveen's 2 exam marks :

80

Enter Praveen's 3 exam marks :

80

Enter Praveen's 4 exam marks :

95

Enter Praveen's 5 exam marks :

55

Enter Praveen's 6 exam marks :

75

Enter Prakash's 1 exam marks :

60

Enter Prakash's 2 exam marks :

50

Enter Prakash's 3 exam marks :

70

Enter Prakash's 4 exam marks :

75

Enter Prakash's 5 exam marks :

55

Enter Prakash's 6 exam marks :

80

Enter Hariharan's 1 exam marks :

40

Enter Hariharan's 2 exam marks :

30

Enter Hariharan's 3 exam marks :

10

Enter Hariharan's 4 exam marks :

45

Enter Hariharan's 5 exam marks :

60

Enter Hariharan's 6 exam marks :

55

Name Marks

1 2 3 4 5 6 Average Deviation

Praveen 45 80 80 95 55 75 71.00 11.67

Prakash 60 50 70 75 55 80 65.25 5.92

Hariharan 40 30 10 45 60 55 41.75 -17.58

Class Average = 59.33

**6. Implement the children's hand game Rock-paper-scissors: Rock Paper Scissors is a two player game. Each player chooses one of rock, paper or scissors, without knowing the other player's choice. The winner is decided by a set of rules: Rock's strength is doubled (temporarily) when fighting scissors, but halved (temporarily) when fighting paper. In the same way, paper has the advantage against rock, and scissors against paper If both players choose the same thing, there is no winner for that round. For this task, the computer will be one of the players. Make 10 rounds of choice, display the score and winner.**

#include<stdio.h>

#include<stdlib.h>

void result(int a[]);

void main()

{

int r=0,p=1,s=2,o,i,b[3]={0,0,0};

int a;

i=0;

while(i<10)

{

printf("Enter 0 for rock\nEnter 1 for paper\nEnter 2 for scissor");

scanf("%d",&o);

if(o==0)

{

srand(time(0));

a=rand()%3;

if(a==0)

{

printf("Draw\n");

b[0]++;

}

else if(a==1)

{

printf("Paper-Computer wins\n");

b[2]++;

}

else if(a==2)

{

printf("Rock-You win\n");

b[1]++;

}

}

else if(o==1)

{

srand(time(0));

a=rand()%3;

if(a==0)

{

printf("Paper-You won\n");

b[1]++;

}

else if(a==1)

{

printf("Draw\n");

b[0]++;

}

else if(a==2)

{

printf("Scissor-Computer wins\n");

b[2]++;

}

}

else if(o==2)

{

srand(time(0));

a=rand()%3;

if(a==0)

{

printf("Rock-Computer won\n");

b[2]++;

}

else if(a==1)

{

printf("Scissor-You won\n");

b[1]++;

}

else if(a==2)

{

printf("Draw\n");

b[0]++;

}

}

else

printf("Invalid input\n");

i++;

}

result(b);

}

void result(int a[])

{

printf("\nGame Over\n");

printf("Computer = %d\n",a[2]);

printf("You = %d\n",a[1]);

if(a[1]>a[2])

printf("You are the winner\n");

else if(a[2]>a[1])

printf("Computer is the winner");

else

printf("Draw");

}

Output:

cseb112@jtl-10:gcc ass36.c –o ass36

cseb112@jtl-10:./ass36

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Paper-You won

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Paper-You won

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Paper-You won

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Enter 0 for rock

Enter 1 for paper

Enter 2 for scissor1

Scissor-Computer wins

Game Over

Computer = 7

You = 3

Computer is the winner