C Programming Codes

Program: Calculate the total number obtained by the student and check that a student is passed or not.

```
#include<stdio.h>

#define MAX_SUB    5  // this is used to restrict the maximum number of subjects
in a class.
#define PASSING_MARKS 180
#define SMARKS 36  // minimum subject marks to pass in per subject.

void main(){

      int  total_number=0, subject_number=0;
      int i, s_count=0;
      // total_number: this will hold the sum of all the subject numbers.
      // subject_number: this is used to hold the inputted value of per subject
number.
      // s_count: it will track the total number of supplementary subjects.
      enum Subjects {Biology=0, Physics, Chemistry, English, Hindi};

      for(i=1; i<=5; i++){

            printf("Enter the number of subject %d:  ", i);
            scanf("%d", &subject_number);

            // Now we need to sum the total_number with the new subject_number.
            total_number += subject_number;

            // checking the subject number that a student got supplementary or
not.

            if(subject_number < SMARKS){
                  // if this condition is satisfied then we need to count a
supplementary subject by one.
                  s_count++;
                  printf("Supplementary in Subject :%d \n\n", i);
            }

      }

      // checking that a student has passed or failed.

      if(total_number >=  PASSING_MARKS){

            if (s_count>=3){
                  printf("\nYou have got supplementary in 3 or more subjects\n
Result: Failed");
            }
            else{
                  printf("\nYou have suplimentry in 1 or 2 subjects\n Result:
Passed");
            }

      }
      else{
            printf("\nFailed\n");
      }
      printf("\n\ntotal Number: %d", total_number);

}
```

==

```
/* Program: in this section we will work on the cylinder area expression
(formula) and we will calculate

 cylinder area formula  = PI*r*r +  2*PI*r*h

 Here r refers to the radius and h refers to the height of the cylinder.

 This is the expression in c to find out the area of the cylinder surface.*/

#include<stdio.h>


#define PI 3.14  // defining a constant PI value using define

void main(){

      float r,h, area;

      printf("Please Enter the radius of a cylinder (dimension in cm): ");
      scanf("%f", &r);
      printf("Please enter the height of the cylinder (dimension in cm): ");
      scanf("%f", &h);

      // Now we need to define the formula of computing the area of a cylinder.
      area = PI*r*r +  2*PI*r*h;

      printf("\n\nCylinder area: %f cm-square", area);

      printf("\n\nCylinder area (in formatted)  %.2f cm-square:", area);

}
```

```
 ===================================================================================

// Program:  check whether an input value is greater than a defined value.
 // in this section we will see the if else statement;
 // if else statement are also called control statement because they control the
 conditions and gives the result according to them

 /* if else statement

       if (condition )
             {
             statements
             }
       else{
             statement
             }
 */
 #include<stdio.h>

 void main(){

       int a=100, b;

       printf("Enter your number: ");
       scanf("%d", &b);

       if (b > a){
             // here we are checking (comparing ) the value of b (input value)
 with a (which is already defined.)
             printf("Input value is greater than 100 < %d", b);
       }
       else{
             // if the above condition does not satisfy then this statement will
 execute.
             printf("Input value is less than 100 > %d",b);
       }
 }


======================================
```

```c
//  Program: in this section we will see the definition of function  and
declaration of functions.

#include<stdio.h>
#include<time.h>

void print_matrix(int arr[3][3]);  // 1. function declaration.


void main(){

// 1. declaration
// 2. definition
// 3. call.

// return-type  function-name  parameter list;
      int matrix[3][3] = {1,2,3,4,5,6,7,8,9};
      int matrixT[3][3] = {{0},{0},{0}};
      int uni_matrix[3][3] = {{1,1,1,},{1,1,1},{1,1,1}};
      int zero_matrix[3][3] = {{0},{0},{0}};
      int i, j;
      print_matrix(matrix); // not return type. but having one argument.

//    printf("Transposed matrix: \n\n");


      for(i=0; i<3; i++){
            // we are running this loop to track each row

            for(j=0; j<3; j++){

                  matrixT[j][i] = matrix[i][j];
            }
      }

      printf("\nTransposed Matrix\n");
      print_matrix(matrixT);

      printf("\nUnit matirx\n");
      print_matrix(uni_matrix);

      printf("\nZero Matrix\n");
      print_matrix(zero_matrix);


}

void print_matrix(int arr[3][3]){
      // 2. function definition.
      int i,j;
      for(i=0; i<3; i++){

            for(j=0; j<3; j++){
                  printf("%d ", arr[i][j]);
            }
            printf("\n");
      }

}
```

==

```
// Program: in this section we will make a program which will display a
matrix and manipulate its value

#include<stdio.h>
#include<time.h>
void main(){


// 2-dimensional arry working as matrix.

      int matrix[3][3] = {1,2,3,4,5,6,7,8,9}; // this is a list of array. {
{1,2,3}, {4,5,6}, {7,8,9} };
      int matrixT[3][3] = {{0}, {0}, {0}};

      // matrix: it is used to hold the original matirx
      // matrixT: it will be used to hold the Transposed of original matirx.

      int i,j;

      printf("Our original matrix: \n\n");

      for(i=0; i<3; i++){
            // we are running this loop to track each row

            for(j=0; j<3; j++){

                  printf("%d ", matrix[i][j]);
            }
            printf("\n");

      }

      printf("Transposed matrix: \n\n");


      for(i=0; i<3; i++){
```

```c
            // we are running this loop to track each row

            for(j=0; j<3; j++){

                  matrixT[j][i] = matrix[i][j];
            }
      }


      for(i=0; i<3; i++){
            // we are running this loop to track each row

            for(j=0; j<3; j++){

                  printf("%d ", matrixT[i][j]);
            }
            printf("\n");
      }


      printf("Making matrixT zero-digonal matrix: ");

      for(i=0; i<3; i++){
            // we are running this loop to track each row

            for(j=0; j<3; j++){

                  if(i==j){
                        matrixT[i][j] = 0;
                        break;
                  }
            }
            printf("\n");
      }


      for(i=0; i<3; i++){
            // we are running this loop to track each row

            for(j=0; j<3; j++){

                  printf("%d ", matrixT[i][j]);
            }
            printf("\n");
      }



}
```

==

```
// Program: Make a program to check the correct pin, if pins are correct
then print "access granted" otherwise print "access denied".

#include<stdio.h>
#define PASSWORD 725257

void main(){

      int input_password;

      printf("Enter the password:");
      scanf("%d", &input_password);

      if(input_password == PASSWORD){

            printf("Access Granted");
      }
      else{
            printf("Access Denied");
      }
}
```

===

```
//Program:  Suppose that we have four girls in marketing to sell three
products, they all have the same three products.

// compute: Total value sales by each girl
// copmpute: Total value of each items sold.
// compute: grand total;

#include<stdio.h>
#include<time.h>
void main(){


// girl1, girl2, girl3, gril4;
// product1, product2, product3

      int Sales[4][3] = { {310,275,365},
                              {210,190,325},
                              {405,235,240},
                              {260,300,380}
                              };
      int Grand_Total = 0;
      int Girls_Sales[4] = {0,0,0,0};
      int Product_Sales[3] = {0,0,0};
      int i,j;
      //---------------------------------------------SALE BY EACH
GIRL----------------------------------------//

      for(i=0; i<4; i++){

            for(j=0; j<3; j++){
                  Girls_Sales[i] += Sales[i][j];
            }
      }


      //---------------------------------------------EACH PRODUCT SALE
----------------------------------------//

      for(i=0; i<3; i++){

            for(j=0; j<4; j++){

                  Product_Sales[i] += Sales[j][i];
            }
      }

      //---------------------------------------------GRAND TOTAL
----------------------------------------//

      for(i=0; i<3; i++){

            Grand_Total += Product_Sales[i];
      }


      printf("\n-----------------------------------TOTAL SALE BY EACH
GIRLS------------------------------------\n");

            for(i=0; i<4; i++){

            printf("Grils %d Sale:  %d \n", i,Girls_Sales[i]);
      }
      printf("\n\n");

      printf("\n-----------------------------------TOTAL SALE OF EACH
PRODUCT------------------------------------\n");
            for(i=0; i<3; i++){

            printf("Product %d Sale: %d \n", i, Product_Sales[i]);
      }
```

```
        printf("\n\n");

        printf("\n----------------------------------GRAND
 TOTAL-----------------------------------\n");

        printf("Grand Total: %d", Grand_Total);

 }
```

```
=================================================================
```