|  |  |  |
| --- | --- | --- |
| **Name:Jacob V Sanoj** | **SRN: PES1UG20EC083** | **Section: F1** |
| **Date: 25-05-2021** | **Week Number: 3** |

|  |  |
| --- | --- |
| **1** | Write a function to reverse a given number and check whether a given number is palindrome or not.Input:Enter the number121Output:The Number 121 is PalindromeInput:Enter the numberOutput:123Number 123 is Not Palindrome |
|  | Program:  #include <stdio.h>  void main()  {  int n;  printf("Enter the number that you want to check if it is a palindrome : ");  scanf("%d", &n);  palindrome(n);  }  void palindrome(int n)  {  int r, q = 0, temp;  temp = n;  while(n != 0)  {  r = n % 10;  n = n/10;  q = q \* 10 + r;  }  printf("The reverse of the number is : %d\n", q);  if(temp == q)  printf("The number is a palindrome\n");  else  printf("The number is not a palindrome\n");  } |
|  | **Output Screenshot:** |
| **2** | Write a C program to compute GCD of three numbers using functions. **Input:**  Enter the values of a,b and c  10 4 16  **Output:**  GCD(10,4,16)=2 |
|  | Program:  #include <stdio.h>  int main()  {  int n1, n2, n3, ans1, ans2;  printf("Enter the values of the digits\n");  scanf("%d%d%d", &n1, &n2, &n3);  ans1 = GCD(n1, n2);  ans2 = GCD(n3, n2);  printf("The GCD of %d, %d and %d = %d\n", n1, n2, n3, ans2);  }  int GCD(int a, int b)  {  while(a != b)  {  if(a > b)  a = a - b;  else  b = b - a;  }  return a;  } |
|  | **Output Screenshot:** |
| **3** | Write a program in C to check Armstrong and perfect numbers using functions.  **Input:**  Input any number: 153  **Output:**  The 153 is an Armstrong number.  The 153 is not a Perfect number.  **Input:**  Input any number: 28  **Output:**  The 28 is not an Armstrong number.  The 28 is a Perfect number. |
|  | Program:  #include <stdio.h>  int armstrong(int n)  {  int digit,sum = 0,num;  num = n;  while(num!=0)  {  digit = num % 10;  sum += digit \* digit \* digit;  num = num / 10;  }  return(n == sum);  }  int perfect(int n)  {  int sum = 0,num;  num=n;  for(int i = 1; i < num; i++)  {  if(num % i == 0)  sum += i;  }  return(n == sum);  }  int main()  {  int n;  printf("Enter the number\n");  scanf("%d", &n);  if(armstrong(n))  printf("The number is an armstrong number\n");  else  printf("The number is not an armstrong number\n");  if(perfect(n))  printf("The number is a perfect number\n");  else  printf("The number is not a perfect number\n");  } |
|  | **Output Screenshot:** |
| **4** | Write a program in C to check whether a number is a prime number or not using function  **Input:**  Input a positive number : 12  **Output:**  The number 12 is not a prime number  **Input:**  Input a positive number : 13  **Output:**  The number 13 is a prime number |
|  | Program:  #include<stdio.h>  int prime1(int n);  int main()  {  int n,prime;  printf("Enter a number : ");  scanf("%d",&n);  prime = prime1(n);  if(prime == 1)  printf("%d is a prime number\n", n);  else  printf("%d is not a prime number\n", n);  }  int prime1(int n)  {  int i = 2;  while(i <= n/2)  {  if(n%i == 0)  return 0;  else  i++;  }  return 1;  } |
|  | **Output Screenshot:** |
| **5** | Write a program in C to convert decimal number to octal number using function  **Input:**  Input any decimal number : 25  **Output:**  Equivalent Octal Number: 17  **Input:**  Input any decimal number : 15  **Output:**  Equivalent Octal Number: 31 |
|  | Program:  #include <stdio.h>  int main()  {  int n, oct;  printf("Enter the value of decimal that you want to convert\n");  scanf("%d", &n);  oct = conversion(n);  printf("The octal of %d is : %d\n", n, oct);  }  int conversion(int n)  {  int oct = 0, temp = 1;  while (n != 0)  {  oct = oct + n%8 + temp;  n = n/8;  temp = temp \* 10;  }  return oct;  } |
|  | **Output Screenshot:** |
| **6** | Write a program in C to find the sum of the series 1!/1+2!/2+3!/3+4!/4+5!/5 using function.  **Output:**  The sum of the series is : 34 |
|  | Program:  #include <stdio.h>  int fact(int n);  int main()  {  int sum = 0;  sum = fact(1)/1 + fact(2)/2 + fact(3)/3 + fact(4)/4 + fact(5)/5;  printf("The sum of the given expression is %d\n", sum);  }  int fact(int n)  {  int sum = 1;  for(int i = 1; i <= n; i++)  sum = sum \* i;  return sum;  } |
|  | **Output Screenshot:** |
| **1** | **Practice Programs**  Write a program to display Fibonacci series in C within a range using a function  **Input:**  Enter range: 5  **Output:**  The fibonacci series is:  0 1 1 2 3 5 |
|  | Program:  #include <stdio.h>  void fibonacci(int range)  {  int a = 0, b = 1, c;  while (a <= range)  {  printf("%d\n", a);  c = a + b;  a = b;  b = c;  }  }  int main()  {  int range;  printf("Enter the range:");  scanf("%d", &range);  printf("The fibonacci series is:\n");  fibonacci(range);  return 0;  } |
|  | **Output Screenshot:** |
| 2 | Write a program to check triangle validity when angles are given using functions.  **Input:**  Enter three angles of triangle:  30  40  60  **Output:**  Triangle is not valid  **Input:**  Enter three angles of triangle:  30  60  90  **Output:**  Triangle is valid |
|  | Program:  #include <stdio.h>  void angles(int a, int b, int c)  {  int angle = a + b + c;  if (angle == 180 && a > 0 && b > 0 && c > 0)  printf("The triangle is valid\n");  else  printf("The triangle is invalid\n");  }  int main()  {  int a, b, c;  printf("Enter the values of angles of triangle\n");  scanf("%d%d%d", &a, &b, &c);  angles(a, b, c);  } |
|  | **Output Screenshot:** |