| **Semester** | **Course Code** | **Course Title** |
| --- | --- | --- |
| Autumn 2024-25 | IT101 | Computer Programming and Problem Solving |

| **Student Details**: Student Name: Rajveer Chaudhari  Roll/Reg No: 202411024  Email: 20241[1024@diu.iiitvadodara.ac.in](mailto:1049@diu.iiitvadodara.ac.in)  Mobile: 6355395811 | |
| --- | --- |
| **Faculty Details**: Faculty Name: Dr. VENKATA PHANIKRISHNA B Department: Computer Science and Engineering  Email: [venkata\_phanikrishna@diu.iiitvadodara.ac.in](mailto:venkata_phanikrishna@diu.iiitvadodara.ac.in) | |
| As.No. | 7 |
| Assessment Title. | Introducing control statements (if-else, while) |
| Date of Submission | 14-November-2024 |

**Recursion Programs**:

* Write a C program to display a multiplication table using recursion.
* Write a C program to find the factorial of a given number using recursion.
* Write a C program to display the Fibonacci series within a given range using recursion.

**Programs on Scope and Storage Classes**

* Write a program to demonstrate variable scope in C, including local, block, global, formal (function parameter), and file scope (using static).
* Write a sample C program to check the scope of the auto storage class.
* Write a C program to find the factorial of a range of numbers using the static storage class.
* Write a C program to demonstrate the use of the extern storage class.
* Write a C program to illustrate the limitations of the register storage class.
* Write a program to compare global variables declared with and without the extern storage class.
* Write a program to restrict reinitialization of a global variable in the local scope.

|  | |
| --- | --- |
| **Question 1** | Write a C program to display a multiplication table using recursion |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a C program to display a multiplication table using recursion.  \*/  #include<stdio.h>  int multi(int x, int y){  if(y == 11){  return 0;  }  else{  printf("%d X %d = %d\n",x,y,x\*y);  y++;  multi(x,y);  }  }  void main(){  int a,b=1;  printf("Enter the number you want mutiplication of: ");  scanf("%d",&a);  multi(a,b);  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your Observation** | Multiplication using recursive |

|  | |
| --- | --- |
| **Question 2** | Write a C program to find the factorial of a given number using recursion. |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a C program to find the factorial of a given number using recursion  \*/  // factourial calculation  #include <stdio.h>  int factorial(int number)  {  if (number == 1 || number == 0)  {  return 1;  }  else  {  return (number \* factorial(number - 1));  }  }  void main()  {  int a;  printf("Enter the number: ");  scanf("%d", &a);  printf("Factorial of %d is %d\n",a, factorial(a));  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your Observation** | Factorial using recursive |

|  | |
| --- | --- |
| **Question 3** | Write a C program to display the Fibonacci series within a given range using recursion. |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a C program to display the Fibonacci series within a given range using recursion.  \*/  #include <stdio.h>  int fibonacci( int p)  {    if (p == 0)  {  return 0;  }  else if (p == 1)  {  return 1;  }  else  {    return fibonacci(p - 1) + fibonacci(p - 2);  }    }  void main()  {  int i, l;  printf("Enter the number to see number of digits in fibonacci series: ");  scanf("%d", &l);  for(int g=0; g<l;g++){  printf("%d ",fibonacci(g));  }  printf("\n");  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your Observation** | Fibonacci using recursive |

|  | |
| --- | --- |
| **Question 4** | Write a program to demonstrate variable scope in C, including local, block, global, formal (function parameter), and file scope (using static). |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a program to demonstrate variable scope in C, including local, block, global, formal (function parameter), and file scope (using static).  \*/  #include<stdio.h>  int b = 24;  static int fsv = 69;  void file(){  printf("The value of file scope variable is %d\n",fsv);  }  void para(int p){  printf("The value of c in formal parameter is %d\n",p);  }  void main(){  int a = 5,c = 68;  printf("The value of a in main function/local scope is %d\n",a);  {  a = 8;  printf("The value of a in block scope is %d\n",a);  }  printf("The value of b in global scope is %d\n",b);  para(c);  printf("The value of file scope variable in main function is %d\n",fsv);  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your Observation** | Local , block ,global scope is used |

|  | |
| --- | --- |
| **Question 5** | Write a sample C program to check the scope of the auto storage class. |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a sample C program to check the scope of the auto storage class.  \*/  #include<stdio.h>  void main(){  auto int a = 5;  printf("value of a using auto int is %d\n",a);  {  auto int a = 76;  printf("value of a using auto int in scope is %d\n",a);  }  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your Observation** | Auto int is used |

|  | |
| --- | --- |
| **Question 6** | Write a C program to find the factorial of a range of numbers using the static storage class. |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a C program to find the factorial of a range of numbers using the static storage class.  \*/  #include<stdio.h>  int fac(int d){  static int l = 1;  if((d == 1) || (d == 0)){  return l;  }  else{  l = l\*d;  return fac(d-1);  }  }  void main(){  int a;  printf("Enter the number for factorial: ");  scanf("%d",&a);  int t = fac(a);  printf("Factorial of %d is %d\n",a,t);  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your Observation** | Factorial using static int |

|  | |
| --- | --- |
| **Question 7** | Write a C program to demonstrate the use of the extern storage class. |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a C program to demonstrate the use of the extern storage class.  \*/  #include<stdio.h>  extern int i;  void main(){  printf("value of i is %d\n",i);  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  }  int i = 532; |
| **Output:** |  |
| **Your Observation** | Extern int is used |

| **Question 8** | Write a program to compare global variables declared with and without the extern storage class. |
| --- | --- |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a C program to illustrate the limitations of the register storage class.  \*/  #include<stdio.h>  void main(){  register int i[10];  for(int j=0; j<10; j++){  printf("a[%d]: ");  scanf("%d", &i[j]);  }  for(int j=0; j<10; j++){  printf("%d ",i[j]);  }  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your**  **Observation** | Showing limitations of register int |

| **Question 9** | Write a program to restrict reinitialization of a global variable in the local scope. |
| --- | --- |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a program to compare global variables declared with and without the extern storage class.  \*/  #include<stdio.h>  #include"ptemp.c"  extern int a;  int i = 45;  void main(){  printf("The value of i displayed withput using extern is %d\n",i);  printf("The value of a displayed with using extern is %d\n",a);  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  } |
| **Output:** |  |
| **Your**  **Observation** | #include”ptemp” is used |

| **Question 10** | Write a program to restrict reinitialization of a global variable in the local scope. |
| --- | --- |
| **Flow chart** |  |
| **Program or Related**  **Content** | /\*  Write a program to restrict reinitialization of a global variable in the local scope.  \*/  #include <stdio.h>  const int a = 12;  int main(){  printf("%d\n", a);  a = 5;  printf("%d\n", a);  printf("Name: Rajveer Chaudhari\n");  printf("Roll Number: 202411024\n");  return 0;  } |
| **Output:** |  |
| **Your**  **Observation** |  |