# Lesson 4 Lab

Task 1: define a program to randomly generate and then print a number in range between 4 to 9

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <time.h>  int main()  {      srand(time(NULL));      int random = rand() % 6;      printf("%d", random + 4);      return 0;  } |

Task 2: define a program to randomly generate and print the top value on a six-sided die (1 ~ 6) for 20 times (Hint: top value is the value on the side facing up)

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <time.h>  int main()  {      srand(time(NULL));      for (int i = 0; i < 20; i++)      {          int random = rand() % 6;          printf("%d ", random + 1);      }      return 0;  } |

Task 3: define a program to calculate 10! and print the result, using recursion [hint: int may not be big enough, you may want to use long]

|  |
| --- |
| #include <stdio.h>  long Factorial(long res)  {      if (res == 0) return 1;      return res \* Factorial(res - 1);  }  int main()  {      long ten = 10;      long result = Factorial(ten);      printf("%ld", result);      return 0;  } |

Task 4: define a program which reads in one positive integer from user, and then print out the input number in reversed order, using recursion

(Eg, when input is 123, output should be 321; when input is 100, output can be 001 or just 1)

|  |
| --- |
| #include <stdio.h>  int Reverser(int num)  {      if (num == 0) return;      printf("%d", num % 10);      Reverser(num / 10);  }  int main()  {      int num;      scanf("%d", &num);      Reverser(num);      return 0;  } |

Task 5: define a program, using recursion, to calculate and print the first 20 numbers in Fibonacci sequence [hint: 0 1 1 2 3 5 8 13 21 34 … (since the third number, each one is the sum of the previous two)]

|  |
| --- |
| #include <stdio.h>  int fib(int n)  {      if (n == 0) return 0;      if (n == 1) return 1;      return fib(n - 1) + fib(n - 2);  }  int main()  {      for (int n = 0; n < 20; n++)      {          printf("%d ", fib(n));      }      return 0;  } |