Lab 3

You are expected to copy and paste your code into each corresponding box in this handout and submit it as a Word document or PDF file before the due. Additionally, your lab instructor will tell you which three questions you must showcase during the lab session. While you may demonstrate your code running in person after the due date, your file must be submitted on time.

Hint #1: For questions asking for 'a function which takes in xyz', the xyz refer to function parameters, not the use of scanf().

Hint #2: For questions asking for 'a program', you are expected to include all necessary #include and define your main function. It’s up to you whether to define additional helper functions, though usually, you don’t need to.

Task 1: Write a **function** which takes in an integer, and then prints the triple of its value (Eg, if input is 2, prints 6)

|  |
| --- |
| void triple(int number) {  printf("%d tripled is %d.\n", number, number \* 3); } |

Task 2: Write a **function** which takes in two integers, and returns the smaller one

|  |
| --- |
| void smallest\_value(int num1, int num2) {  if (num1 > num2) {  printf("%d is the smaller value.\n", num2);  } else if (num1 < num2) {  printf("%d is the smaller value.\n", num1);  } else {  printf("They are the same.");  } } |

Task 3: Write a **function** which takes in a float, prints it, and then return the absolute value of it.

|  |
| --- |
| float absolute\_value(float number) {  printf("You entered: %f\n", number);  if (number < 0) {  return -number;  }  return number; } |

Task 4: Write a **function** which takes in a char, and prints the input char 100 times

|  |
| --- |
| void char\_times\_100(char input) {  for (size\_t count = 0; count < 100; count++) {  if (count % 20 == 0) {  printf("\n");  }  printf("%c", input);  } } |

Task 5: Write a **function** which takes in two doubles, and prints the non-negative difference (Eg, the bigger one – the smaller one) [Hint: if the two inputs are the same, prints 0]

|  |
| --- |
| void non\_negative\_difference(double num1, double num2) {  if (num1 > num2) {  printf("The difference is %lf.", num1 - num2);  } else if (num1 < num2) {  printf("The difference is %lf.", num2 - num1);  } else {  printf("0");  } } |

Task 6: define a function which takes in two integers, and then prints the result of the first number got divided by the second one as a double (Eg, if input 1 4, should print 0.25)

|  |
| --- |
| void divided\_by(int num1, int num2) {  double result = (double) num1 / (double) num2;  printf("%d divided by %d is %f.", num1, num2, result); } |

Task 7: define a **program** which reads in a double, and then prints it out as three times: (1) As an int (2) as a double (3) as a float

|  |
| --- |
| #include <stdio.h>  int main() {  double input;  printf("Enter a double value:\n");  scanf("%lf", &input);   int integer\_value = (int) input;  double double\_value = input;  float float\_value = (float) input;   printf("%lf as an integer: %d\n", input, integer\_value);  printf("%lf as a double: %lf\n", input, double\_value);  printf("%lf as a float: %f", input, float\_value);   return 0; } |

**You are required to use function from math library to answer Task 8 ~ 10:**

Task 8: define a **program** which reads in two doubles, and then prints out the one which has bigger absolute value

|  |
| --- |
| #include <stdio.h> #include <math.h>  int main() {  double num1, num2;  printf("Enter two double values separated by a space:\n");  scanf("%lf %lf", &num1, &num2);   double abs\_num1 = fabs(num1);  double abs\_num2 = fabs(num2);   if (abs\_num1 > abs\_num2) {  printf("%lf is the larger absolute value.", abs\_num1);  } else if (abs\_num1 < abs\_num2){  printf("%lf is the larger absolute value.", abs\_num2);  } else {  printf("They have the same absolute value.");  }   return 0; } |

Task 9: define a **program** which reads in one positive float (called x), and then prints the value of sin(2x) [Hint: sin(2x) = 2 sin(x) \*cos(x)]

|  |
| --- |
| #include <stdio.h> #include <math.h>  int main() {  float input;  printf("Enter a positive float value to calculate 2sin(x):\n");  scanf("%f", &input);   float result = 2 \* sin(input) \* cos(input);  printf("The result of sin(2 \* %f) is %f", input, result);   return 0; } |

Task 10: define a **program** which reads in two positive integers, and then prints their log (base 10) of their sum [Assumption: the user will type in two positive integers]

|  |
| --- |
| #include <stdio.h> #include <math.h>  int main() {  int num1, num2;  printf("Enter two positive integers separated by a space:\n");  scanf("%d %d", &num1, &num2);   double result = log(num1 + num2);  printf("The log of (%d + %d) is %lf.", num1, num2, result);   return 0; } |

Task 11: 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() {  // Seed the random number generator  srand(time(NULL));   int result = (rand() % 6) + 4;   printf("The random result is: %d", result);   return 0; } |

Task 12: 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() {  // Seed the random number generator  srand(time(NULL));   for (size\_t roll = 0; roll < 20; roll++) {  int die\_value = (rand() % 6) + 1;   printf("The result of roll #%llu is: %d\n", roll + 1, die\_value);  }  return 0; } |

Task 13: 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(int number) {  // Set the base case  if (number < 2) {  return 1; // 0! and 1! are both 1  }  return number \* factorial(number - 1); }  int main() {  int number = 10;  long result = factorial(number);  printf("%ld", result);   return 0; } |

Task 14: 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>  void reverse\_number(int input) {  // Set the base case  if (input == 0) {  return; // End recursion  }  printf("%d", input % 10);  reverse\_number(input / 10);  }  int main() {  for (int nth\_term = 1; nth\_term <= 20; nth\_term++) {  int result = fibonacci(nth\_term);  printf("Term #%d in the Fibonacci Sequence: %d\n", nth\_term, result);  }  return 0; } |

Task 15: 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 fibonacci(int nth\_term) {  // Set the base case  if (nth\_term < 2) {  return nth\_term;  }  return fibonacci(nth\_term -1) + fibonacci(nth\_term - 2); }  int main() {  int nth\_term = 20;  int result = fibonacci(nth\_term);  printf("Term #%d in the Fibonacci Sequence: %d", nth\_term, result);  return 0; } |