

C Language Programs

Assignment 3

Submitted By:

Amitanshu Kumar Swain

1CSE 2Y Enrolment No.A2305220070

INDEX

|  |  |
| --- | --- |
| Experiment 1 | Write a C program to find power of a number using for loop. |
| Experiment 2 | Write a C program to find all factors of a number |
| Experiment 3 | Write a C program to find HCF (GCD) of two numbers. |
| Experiment 4 | Write a C program to find LCM of two numbers. |
| Experiment 5 | Write a C program to find all prime factors of a number. |
| Experiment 6 | Write a C program to check whether a number is Strong number or not. |
| Experiment 7 | Write a C program to print all Strong numbers between 1 to n. |
| Experiment 8 | Write a C program to convert Hexadecimal to Decimal number system. |
| Experiment 9 | Write a C program to input week number and print weekday. |
| Experiment 10 | Write a C program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer.  Calculate percentage and grade according to following:  Percentage >= 90% : Grade A  Percentage >= 80% : Grade B  Percentage >= 70% : Grade C  Percentage >= 60% : Grade D  Percentage >= 40% : Grade E  Percentage < 40% : Grade F. |

Experiment 1

Objective: Write a C program to find power of a number using for loop.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int base, exponent;**

**long long power = 1;**

**int i;**

**printf("Enter base: ");**

**scanf("%d", &base);**

**printf("Enter exponent: ");**

**scanf("%d", &exponent);**

**for(i=1; i<=exponent; i++)**

**{**

**power = power \* base; }**

**printf("%d ^ %d = %lld", base, exponent, power);**

**return 0;**

**}**

Output:

Enter base:

Enter exponent:

Experiment 2

Objective: Write a C program to find all factors of a number.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int i, num;**

**printf("Enter any number to find its factor: ");**

**scanf("%d", &num);**

**printf("All factors of %d are: \n", num);**

**for(i=1; i<=num; i++)**

**{**

**if(num % i == 0)**

**{**

**printf("%d, ",i);**

**}**

**}**

**return 0;**

**}**

Output:

Enter the number to find its factor:

All factors of \_\_ are:

Experiment 3

Objective: Write a C program to find HCF (GCD) of two numbers.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int i, num1, num2, min, hcf=1;**

**printf("Enter any two numbers to find HCF: ");**

**scanf("%d%d", &num1, &num2);**

**min = (num1<num2) ? num1 : num2;**

**for(i=1; i<=min; i++)**

**{**

**if(num1%i==0 && num2%i==0)**

**{**

**hcf = i;**

**}**

**}**

**printf("HCF of %d and %d = %d\n", num1, num2, hcf);**

**return 0;**

**}**

Output:

Enter the two numbers to find HCF:

Experiment 4

Objective: Write a C program to find LCM of two numbers.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int i, num1, num2, max, lcm=1;**

**printf("Enter any two numbers to find LCM: ");**

**scanf("%d%d", &num1, &num2);**

**max = (num1 > num2) ? num1 : num2;**

**i = max;**

**while(1)**

**{**

**if(i%num1==0 && i%num2==0)**

**{**

**lcm=i;**

**break;**

**}**

**i += max;**

**}**

**printf("LCM of %d and %d = %d", num1, num2, lcm);**

**return 0;**

**}**

Output:

Enter the two numbers to find LCM:

LCM of\_\_ and \_\_ =

Experiment 5

Objective: Write a C program to find all prime factors of a number.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int i, j, num, isPrime;**

**printf("Enter any number to print Prime factors: ");**

**scanf("%d", &num);**

**printf("All Prime Factors of %d are: \n", num);**

**for(i=2; i<=num; i++)**

**{**

**if(num%i==0)**

**{**

**isPrime = 1;**

**for(j=2; j<=i/2; j++)**

**{**

**if(i%j==0)**

**{**

**isPrime = 0;**

**break;**

**}**

**}**

**if(isPrime==1)**

**{**

**printf("%d, ", i);**

**}**

**}**

**}**

**return 0;}**

Output:

Enter the number to print Prime factors:

All Prime factors of \_\_ are:

Experiment 6

Objective: Write a C program to check whether a number is Strong number or not.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int i, originalNum, num, lastDigit, sum;**

**long fact;**

**printf("Enter any number to check Strong number: ");**

**scanf("%d", &num);**

**originalNum = num;**

**sum = 0;**

**while(num > 0)**

**{**

**lastDigit = num % 10;**

**fact = 1;**

**for(i=1; i<=lastDigit; i++)**

**{**

**fact = fact \* i;**

**}**

**sum = sum + fact;**

**num = num / 10;**

**}**

**if(sum == originalNum)**

**{**

**printf("%d is STRONG NUMBER", originalNum);**

**}**

**else**

**{**

**printf("%d is NOT STRONG NUMBER", originalNum);**

**}**

**return 0;**

**}**

Output:

Enter any number to check Strong number:

Experiment 7

Objective: Write a C program to print all Strong numbers between 1 to n.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int i, j, cur, lastDigit, end;**

**long long fact, sum;**

**printf("Enter upper limit: ");**

**scanf("%d", &end);**

**printf("All Strong numbers between 1 to %d are:\n", end);**

**for(i=1; i<=end; i++)**

**{**

**cur = i;**

**sum = 0;**

**while(cur > 0)**

**{**

**fact = 1ll;**

**lastDigit = cur % 10;**

**for( j=1; j<=lastDigit; j++)**

**{**

**fact = fact \* j;**

**}**

**sum += fact;**

**cur /= 10;**

**}**

**if(sum == i)**

**{**

**printf("%d, ", i);**

**}**

**}return 0;**

**}**

Output:

Enter upper limit:

All Strong numbers between 1 and \_\_ are:

Experiment 8

Objective: Write a C program to convert Hexadecimal to Decimal number system.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**#include <math.h>**

**#include <string.h>**

**int main()**

**{**

**char hex[17];**

**long long decimal, place;**

**int i = 0, val, len;**

**decimal = 0;**

**place = 1;**

**printf("Enter any hexadecimal number: ");**

**gets(hex);**

**len = strlen(hex);**

**len--;**

**for(i=0; hex[i]!='\0'; i++)**

**{**

**if(hex[i]>='0' && hex[i]<='9')**

**{**

**val = hex[i] - 48;**

**}**

**else if(hex[i]>='a' && hex[i]<='f')**

**{**

**val = hex[i] - 97 + 10;**

**}**

**else if(hex[i]>='A' && hex[i]<='F')**

**{**

**val = hex[i] - 65 + 10;**

**}**

**decimal += val \* pow(16, len);**

**len--;**

**}**

**printf("Hexadecimal number = %s\n", hex);**

**printf("Decimal number = %lld", decimal);**

**return 0;**

**}**

Output:

Enter any Hexadecimal number:

Hexadecimal number:

Decimal number:

Experiment 9

Objective: Write a C program to input week number and print weekday.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int week;**

**printf("Enter week number (1-7): ");**

**scanf("%d", &week);**

**if(week == 1)**

**{**

**printf("Monday");**

**}**

**else if(week == 2)**

**{**

**printf("Tuesday");**

**}**

**else if(week == 3)**

**{**

**printf("Wednesday");**

**}**

**else if(week == 4)**

**{**

**printf("Thursday");**

**}**

**else if(week == 5)**

**{**

**printf("Friday");**

**}**

**else if(week == 6)**

**{**

**printf("Saturday");**

**}**

**else if(week == 7)**

**{**

**printf("Sunday");**

**}**

**else**

**{**

**printf("Invalid Input! Please enter week number between 1-7.");**

**}**

**return 0;**

**}**

Output:

Enter week number(1-7):

Experiment 10

Objective: Write a C program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer.

Calculate percentage and grade according to following:

Percentage >= 90% : Grade A

Percentage >= 80% : Grade B

Percentage >= 70% : Grade C

Percentage >= 60% : Grade D

Percentage >= 40% : Grade E

Percentage < 40% : Grade F.

Language Used: C

Theory: The compiler is a type of language processor which converts a high level program into machine level program at one go rather than line by line. The language processor converts all user instructions into machine understandable language.

Program:

**#include <stdio.h>**

**int main()**

**{**

**int phy, chem, bio, math, comp;**

**float per;**

**printf("Enter five subjects marks: ");**

**scanf("%d%d%d%d%d", &phy, &chem, &bio, &math, &comp);**

**per = (phy + chem + bio + math + comp) / 5.0;**

**printf("Percentage = %.2f\n", per);**

**if(per >= 90)**

**{**

**printf("Grade A");**

**}**

**else if(per >= 80)**

**{**

**printf("Grade B");**

**}**

**else if(per >= 70)**

**{**

**printf("Grade C");**

**}**

**else if(per >= 60)**

**{**

**printf("Grade D");**

**}**

**else if(per >= 40)**

**{**

**printf("Grade E");**

**}**

**else**

**{**

**printf("Grade F");**

**}**

**return 0;**

**}**

Output:

Enter five subject marks:

Percentage:

Grade: