* Assignment - PSTREE

#include<stdio.h>

void main()

{

system("pstree");

}

Assignment - question 1

#include<stdio.h>

int main()

{

int a,b,c;

printf("ENTER THREE INTEGERS\n");

scanf("%d%d%d",&a,&b,&c);

if((a==b) && (b==c) && (c==a))

printf("0\n");

else

printf("0\n");

return 0;

}

Assignment – question 2

#include<stdio.h>

int main()

{

int M,N;

printf("ENTER TWO INTEGERS\n");

scanf("%d%d",&M,&N);

if(M%N!=0)

printf("0\n");

else

printf("M/N\n");

return 0;

}

Assignment – question 3

#include<stdio.h>

int main()

{

float a,b,c;

printf("ENTER THE THREE SIDES OF A TRIANGLE\n");

scanf("%f%f%f",&a,&b,&c);

if((a+b>c) && (b+c>a) && (c+a>b))

printf("1\n");

else

printf("0\n");

return 0;

}

Assignment – question 4

#include <stdio.h>

int main()

{

int array[100], search, c, n;

printf("Enter the number of elements in array\n");

scanf("%d", &n);

printf("Enter %d integer(s)\n", n);

for (c = 0; c < n; c++)

scanf("%d", &array[c]);

printf("Enter a number to search\n");

scanf("%d", &search);

for (c = 0; c < n; c++)

{

if (array[c] == search) /\* If required element is found \*/

{

printf("%d is present at location %d.\n", search, c+1);

break;

}

}

if (c == n)

printf("%d isn't present in the array.\n", search);

return 0;

}

Assignment – On roman numerals

#include<stdio.h>

#include<string.h>

int digit(char);

int main(){

char roman\_Number[1000];

int i=0;

long int number =0;

printf("Enter any roman number (Valid digits are I, V, X, L, C, D, M): \n");

scanf("%s",roman\_Number);

while(roman\_Number[i]){

if(digit(roman\_Number[i]) < 0){

printf("Invalid roman digit : %c",roman\_Number[i]);

return 0;

}

if((strlen(roman\_Number) -i) > 2){

if(digit(roman\_Number[i]) < digit(roman\_Number[i+2])){

printf("Invalid roman number");

return 0;

}

}

if(digit(roman\_Number[i]) >= digit(roman\_Number[i+1]))

number = number + digit(roman\_Number[i]);

else{

number = number + (digit(roman\_Number[i+1]) - digit(roman\_Number[i]));

i++;

}

i++;

}

printf("Its decimal value is : %ld",number);

return 0;

}

int digit(char c){

int value=0;

switch(c){

case 'I': value = 1; break;

case 'V': value = 5; break;

case 'X': value = 10; break;

case 'L': value = 50; break;

case 'C': value = 100; break;

case 'D': value = 500; break;

case 'M': value = 1000; break;

case '\0': value = 0; break;

default: value = -1;

}

return value;

}

Assignment – ipv4

#include<stdio.h>

void main()

{

int a,b,c,d;

printf("An IPV4 adress contains 4 parts");

printf("\nexample: 123.456.789.012");

printf("\nas in the upper example \n1st part is:123 \n2nd part is:456 \n3rd part is:789 \n4th part is:012");

printf("\nnow you enter 1st part of IPV4 for checking:");

scanf("%d",&a);

printf("enter 2nd part:");

scanf("%d",&b);

printf("enter 3rd part:");

scanf("%d",&c);

printf("enter 4th part:");

scanf("%d",&d);

if(a>=0 && b>=0 && c>=0 && d>=0 && a<=255 && b<=255 && c<=255 && d<=255)

{

printf("\nThis means %d.%d.%d.%d is a valid IPV4",a,b,c,d);

}

else

{

printf("%d.%d.%d.%d is not a valid IPV4",a,b,c,d);

}

}