|  |
| --- |
| Armghan Ahmad |
| OOP Lab-08: Lab work & Home Work(Task) |
| 20P-0183 |

|  |
| --- |
| BSCS 2B |

Question 1:

#include<iostream>

using namespace std;

int main()

{

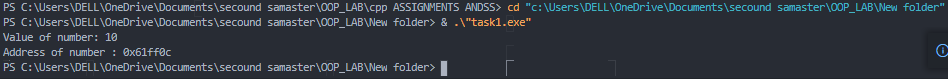
int num=10;

cout << "Value of number: "<< num << endl;

cout << "Address of number : " << &num << endl;

}

Output:



Question 2:

#include <iostream>

using namespace std;

int main()

{

int var1=3; int var2 = 24;

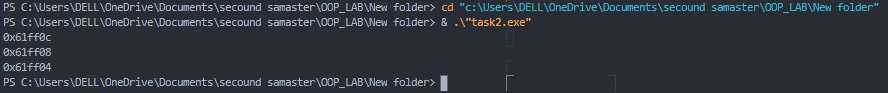
int var3=17;

cout << &var1 << endl;

cout << &var2 << endl; cout << &var3 << endl;

}

Output:



Question 3:

#include<iostream>

using namespace std;

int main()

{

int num=10;

int \*ptr;

ptr=&num;

cout << "num="<< num << endl;

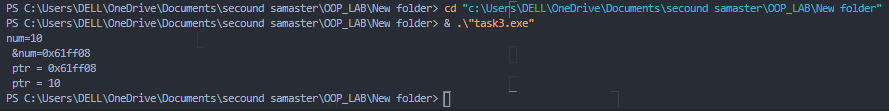
cout << " &num="<< &num << endl;

cout <<" ptr = " << ptr << endl;

cout<<" ptr = " << \*ptr << endl;

}

Output:



Question 4:

#include<iostream>

using namespace std;

int main()

{

int \*pc, c;

c = 5;

cout << "Address of c (&c): " << &c << endl;

cout << "Value of c (c): "<<c<<endl<< endl;

pc = &c;*// Pointer pc holds the memory address of variable c cout << "Address that pointer pc holds (pc): "<< pc << endl;*

cout << "Content of the address pointer pc holds (pc): "<< \*pc << endl<< endl; c=11;*// The content inside memory address &c is changed from 5 to 11.*

cout << "Address pointer pc holds (pc): " << pc << endl;

cout << "Content of the address pointer pc holds (pc): " << \*pc << endl<< endl;

\*pc = 2;

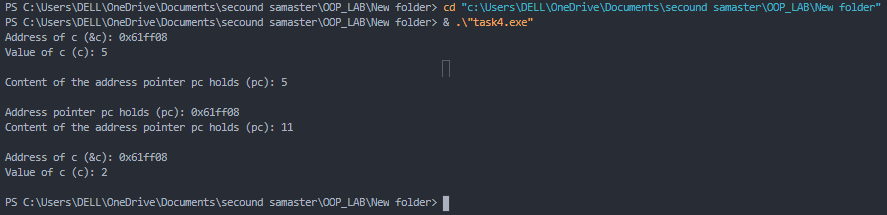
cout << "Address of c (&c): " << &c << endl;

cout << "Value of c (c): "<<c<< endl << endl;

return 0;

}

Output:



Question 5:

#include <iostream>

using namespace std; int main()

{

int \*ptr, x; x = 10;

ptr = &x;

cout << "The value of x: \t" <<x<< endl;

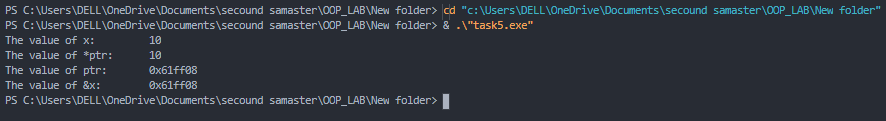
cout << "The value of \*ptr: \t" << \*ptr << endl;

cout << "The value of ptr: \t" <<ptr << endl;

cout << "The value of &x: \t" <<&x << endl;

}

Output:



Question 6:

#include <iostream>

using namespace std;

int main ()

{

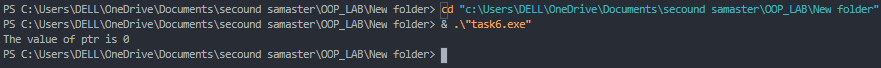
int ptr = NULL;

cout << "The value of ptr is " << ptr ;

return 0;

}

Output:



Question 7:

#include <iostream>

using namespace std;

int main()

{

void \*ptr1, \*ptr2, \*ptr3, \*ptr4;

 int i; char c; float f; double d;

cout<<"Enter the integer value : "<<endl;

cin>>i;

cout<<"Enter the character: "<<endl;

 cin>>c;

cout<<"Enter the float value: "<<endl;

cin>>f;

cout<<"Enter the double value: "<<endl;

cin>>d;

ptr1=&i;

ptr2=&c;

ptr3=&f;

ptr4=&d;

cout<<"Your entered integer address is: "<<ptr1<<endl;

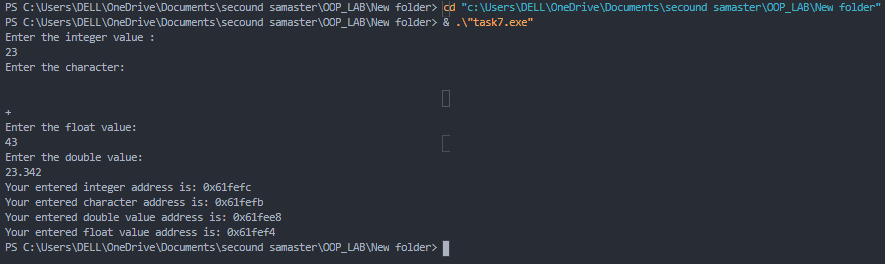
cout<<"Your entered character address is: "<<ptr2<<endl;

cout<<"Your entered double value address is: "<<ptr4<<endl;

cout<<"Your entered float value address is: "<<ptr3<<endl;

return 0;

}

Output: 

Question 8:

#include <iostream>

using namespace std;

int main()

{

int value = 7;

int \*ptr = &value;

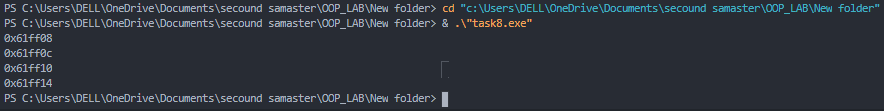
cout << ptr << '\n';

cout << ptr+1 << '\n'; cout << ptr+2 << '\n'; cout << ptr+3 << '\n';

return 0;

}

Output:



Question 9:

#include <iostream>

using namespace std;

int main()

{

int a = 10, \*b, \*c;

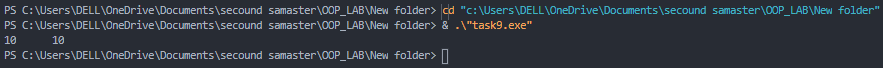
b = &a;

c = b;

cout << \*b << "\t" << \*c;

}

Output:



Question 10:

#include<iostream>

using namespace std;

int main() {

int b[3];

int\* a = b;

cout<<a<<endl;

a++;

cout<<a<<endl;

a;

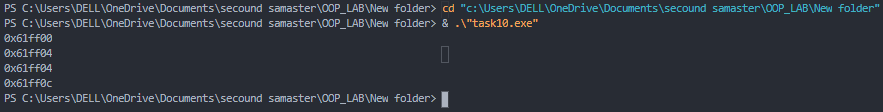
cout<< a <<endl;

a = a + 2;

cout<<a<<endl;

}

Output:

Question 11:

#include<iostream>

using namespace std;

void fun(int, int );

int main(){

int A=10,B=20;

cout << "\nValues before calling";

cout << "\nA: " << A;

cout << "\nB:"<< B;

fun(A,B);

*//Statement 1*

cout << "\nValues after calling";

cout << "\nA: " << A;

cout << "\nB:"<< B;

}

void fun(int *X*, int *Y*) {

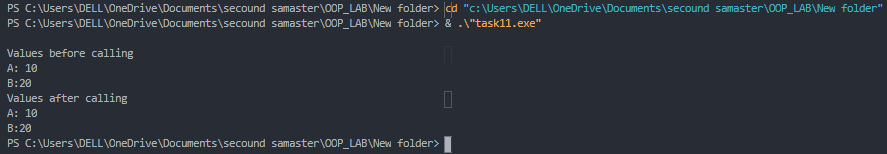
*//Statement 2*

*X*=11;

*Y*=22;

}

Output:



Question 12:

#include<iostream>

using namespace std;

void fun(int\*, int\*);

int main(){

int A=10,B=20;

cout << "\nValues before calling";

cout << "\nA: " << A;

cout << "\nB:"<< B;

fun(&A,&B);

*//Statement 1*

cout << "\nValues after calling";

cout << "\nA: " << A;

cout << "\nB: " << B;

}

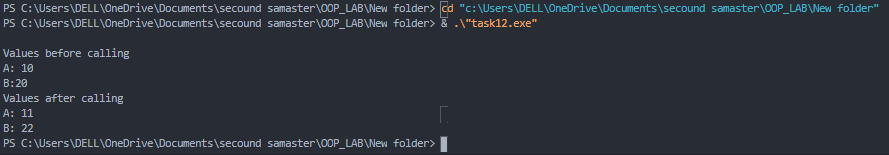
void fun(int \**X*, int \**Y*) {*//Statement 2*

\**X*=11;

\**Y*=22;

}

Output:

Question 13:

#include<iostream>

using namespace std;

int \*reference(int);

int main() {

int A=10;

int \*ptr;

cout << "\nAddress of " << A ;

cout<<" in main() is " << &A;

ptr = reference(A);

cout << "\nAddress of " << A ; cout<<" in reference() was "<<ptr;

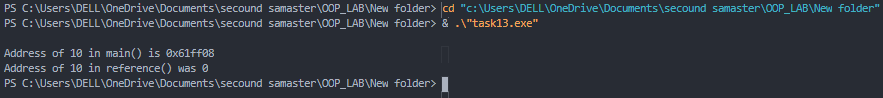
}

int \*reference(int *n*) {

return &*n*;

}

Output:



Question 14:

#include<iostream>

using namespace std;

int Sum(int, int);

int (\*ptr)(int,int);

int main() {

int a, b, rs;

cout << "\nEnter 1st number: ";

cin >> a;

cout << "\nEnter 2nd number: ";

cin >> b;

ptr = Sum;

rs= (\*ptr)(a,b);

cout << "\nThe sum is : " << rs;

}

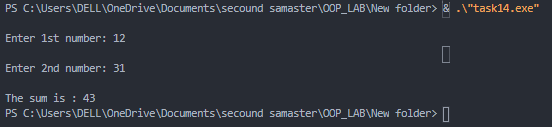
int Sum(int *x*, int *y*)

{

return *x*+*y*;

}

Output:



Question 15:

#include <iostream>

using namespace std;

int main()

{

short numbers[] = {10, 20, 30, 40, 50};

cout << "The first element of the array is ";

cout << \*numbers << endl;

return 0;

}

Output:



Question 16:

#include <iostream>

using namespace std;

int main()

{

int array[5] = {2, 4, 6, 8, 10};

int \*aptr;

aptr = array;

cout << "Address of Array:\t" << &array[0] << endl;

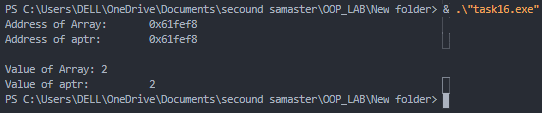
cout << "Address of aptr:\t" << aptr << endl;

cout << "\nValue of Array:\t" << array[0] << endl;

cout << "Value of aptr:\t\t" << \*aptr << endl;

}

Output:



Question 17:

#include<iostream>

using namespace std;

int main()

{

int y[10];

int \*

yptr;

yptr=y;

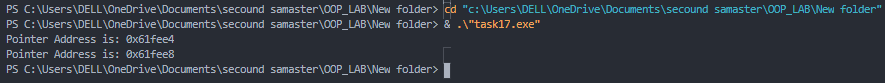
cout <<"Pointer Address is: " <<yptr<<endl;

yptr++;

cout <<"Pointer Address is: " <<yptr;

}

Output:



Question 18:

#include<iostream>

using namespace std;

int main()

{

int array[5] = {2, 4, 6, 8, 10};

int \*aptr;

aptr = array;

cout << "Address of Array:\t" << &array[0] << endl;

cout << "Address of aptr:\t" << aptr << endl;

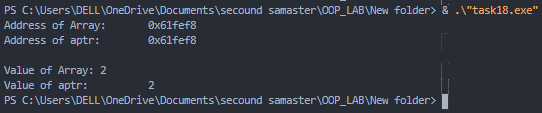
cout << "\nValue of Array:\t" << array[0] << endl;

cout << "Value of aptr:\t\t" << \*aptr << endl;

return 0;

}

Output:



Question 19:

#include<iostream>

using namespace std;

int main() {

    float a[5];

    float \*ptr; cout << "Displaying address using arrays: "<<endl;

    for (int i = 0; i < 5; ++i)

    {

        cout << "&a[" << i << "] = " << &a[i] <<endl;

    }

    ptr = a;*// ptr = &a[0] cout<<"\nDisplaying address using pointers: "<<endl;*

    for (int i = 0; i < 5; ++i)

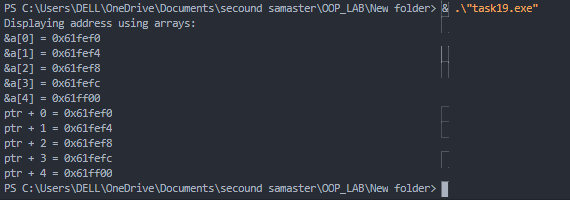
    {

        cout << "ptr + " << i << " = "<<ptr+i <<endl;

    }

}

OUTPUT:



Question 20:

#include<iostream>

using namespace std;

int main() {

int array []={78,45,12,89,56,23,79,46,13,82};

int \*ptr;*// Pointer variable*

*// Assigning reference of array in pointer variable*

ptr = array;

cout << "\nValues: ";

for(int a=1;a<=10;a++)

{

cout <<\*ptr<<", ";*// Displaying values of array*

*// using pointer Incrementing pointer variable*

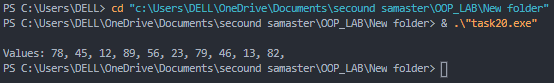
ptr++;

}

return 0;

}

Output:



Question 21:

#include<iostream>

using namespace std;

int main() {

int x=10,y=20,z=30;

*// Declaring array of three pointer*

int \*array[3];

*// Assigning reference of x to array Oth position*

array[0] = &x;

*// Assigning reference of y to array 1th position*

array[1] =&y;

*// Assigning reference of z to array 2nd position*

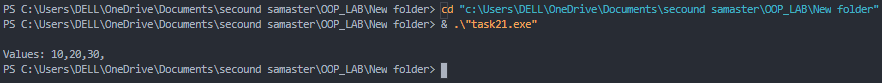
array[2] = &z;

cout << "\nValues: "; for(int a=0;a<3;a++)

cout << \*array[a]<<",";

}

Output:



Question 22:

#include<iostream>

using namespace std;

int main() {

char\* a="Hello";

cout<<a<<endl;

cout<<&a<<endl;

cout<<\*a<<endl;

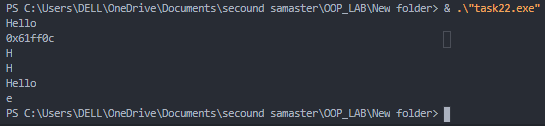
cout<<a[0]<<endl;

cout<<a++<<endl;

cout<<\* a++<<endl;

}

Output:



Question 23:

#include<iostream>

using namespace std;

int main()

{

int a = 10;

cout << "The address of \"a\" variable is " << &a << endl;

*//pointer to int*

int \*p;

p = &a;*// now p is equal to the address of 'a' variable*

cout << "The value of \"p\" variable is " << p << endl;

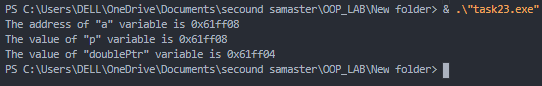
*//pointer to pointer*

int\*\* doublePtr=&p;

 cout << "The value of \"doublePtr\" variable is " << doublePtr << endl;

}

Output:



Question 24:

#include<iostream>

using namespace std;

int main()

{

int a = 10;

cout << "The address of \"a\" variable is " << &a << endl;

*//pointer to int*

int\* p;

p=&a;*// now p is equal to the address of 'a' variable cout << "The value of \"p\" variable is " << p << endl;*

*//pointer to pointer*

int\*\* doublePtr=&p;

cout << "The value of \"doublePtr\" variable is " << doublePtr << endl;

cout<< "The value is: "<< a<<endl;

}

Output:

