**Part-3**

1. What is a subscripted value? -🡪 Array index
2. How to declare 1d, 2d, 3d array?
3. What is output?

int main()

{

char i=300;

printf("%d",(int) i); //ans=44

return 0;

}

1. What is output?

int main()

{

unsigned char i=255;

printf("%d",i); //ans=255

return 0;

}

1. What is the output?

int main()

{

int c[2][2],i,j;

printf("Enter 4 numbers:\n");

for(i=0;i<2;++i)

{

for(j=0;j<2;++j)

{

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

}

}

Function(c); //pass array base addr

return 0;

}

void Function(int c[][3]) //🡪cant exceed limit

{

int i,j;

printf("Displaying:\n");

for(i=0;i<2;++i)

for(j=0;j<2;++j)

printf("%d\n",c[i][j]);

}

1. What is output?

int main()

{

unsigned int i=1,c=0;

while(i)

{

i<<=1; c++; // i=i<<1 ; c=8

}

printf("%d",c/8); //ans=1

return 0;

}

1. What is output?

int main()

{

int a[] = {6,7,5,3,1,4,2,10,9};

int s = \*(&a + 1) –a; //both are same 🡪(int\*)(&a + 1) – (int\*)a;

printf(“%d”,s); //ans=9

}

1. What is output

int main()

{

int a[] = {6,7,5,3,1,4,2,10,9};

int s = (char\*)(&a + 1)- (char\*)a;

printf("%d",s) ;//36

}

1. How to find data type(int,char) size without using sizeof operator? ref qus6
2. How to find int array size without using sizeof operator? ref qus8
3. How to find array lengh without using sizeof operator? ref qus7
4. Write macro to find lengh of array
5. Write macro to find sizeof of array
6. Size of struct?

struct a

{

int a;

int b;

} sData[2];

int main()

{

struct a size;

printf("\nSize of Structure : %d Bytes",sizeof(size)); //ans=8

return 0;

}

1. Size of struct array?

struct a

{

int a;

int b;

} sData[2];

int main()

{

printf("\nSize of Structure : %d Bytes",sizeof(sData)); //ans=16

return 0;

}

1. What is ouput

struct

{

int a;

char b;

} sData[2]; // e.g addr loc: 0 4 (oth index) 8 12 (1st index) 16

int main()

{

int struct\_size=(char\*)(&sData[1].a)-(char\*)(&sData[0].a); //8

int struct\_arr\_size=(char\*)(&sData+1)-(char\*)(sData); //16

printf("\nSize of Structure : %d Bytes",struct\_size); //ans=8

printf("\nSize of Structure : %d Bytes",struct\_arr\_size); //ans=16

return 0;

}

1. How to find structure size without using sizeof operator?
2. Is sizeof for a struct equal to the sum of sizeof of each member? 🡪no
3. What is output?

***#include <stdio.h>***

***#define message\_for(a, b) \***

***printf(#a " and " #b ": We love you!\n")***

***int main(void) {***

***message\_for(Carole, Debra);***

***return 0;***

***}***

1. Stringize (#) Operator?
2. What is output?

***#include <stdio.h>***

***#define tokenpaster(n) printf ("token" #n " = %d", token##n)***

***int main(void) {***

***int token34 = 40;***

***tokenpaster(34);***

***return 0;***

***}***

1. Token Pasting (##) Operator?
2. What is output?

***#include <stdio.h>***

***#if !defined (MESSAGE)***

***#define MESSAGE "You wish!"***

***#endif***

***int main(void) {***

***printf("Here is the message: %s\n", MESSAGE);***

***return 0;***

***}***

1. Defined() Operator?
2. Write a program for bit position swap? // data=5-0000 0101 swap(data,0,7) =132(ans)
3. Write a program for bit reverse? Rev of 5 : 1010 0000 🡺ans= (in dec)

Tricks/hacks

&a => This is the pointer to array which points at the same memory address as a.

&a + 1 => It points at the address after the end of the array.

\*(a+1) => Dereferencing to \*(&a + 1) gives the address after the end of the last element.

\*(a+1)-a => Subtract the pointer to the first element to get the length of the array.

Sizeof 🡪 # define my\_sizeof(type) ((char \*)(&type+1)-(char\*)(&type))

Length 🡪 # define my\_length(a) ((int\*)(&a + 1)- (int\*)a)