1.What will be output if you will compile and execute the following c code?

void main(){

   int i=320;

   char \*ptr=(char \*)&i;

   printf("%d",\*ptr);

}

Options :

A)360

B)120

C)64

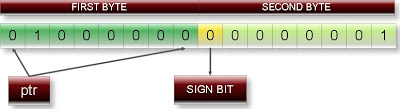
D)Compilation error

ANS : C

Explanation :

As we know size of int data type is two byte while char pointer can pointer one byte at time.

Memory representation of int i=320



So char pointer ptr is pointing to only first byte as shown above figure.

\*ptr i.e. content of first byte is 01000000 and its decimal value is 64.

2. What will be output if you will compile and execute the following c code?

void main(){

char c=125;

    c=c+10;

    printf("%d",c);

}

A)135

B)-121

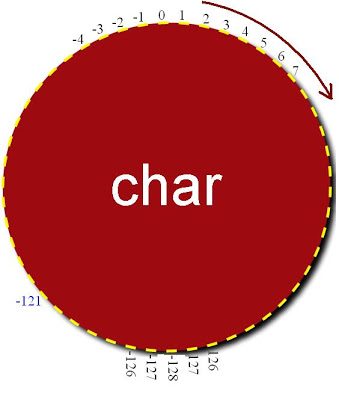
C)+INF

D)Complilation Error

ANS:B

Explanations:

As we know char data type shows cyclic properties i.e. if you will increase or decrease the char variables beyond its maximum or minimum value respectively it will repeat same value according to following cyclic order:



So,

125+1= 126

125+2= 127

125+3=-128

125+4=-127

125+5=-126

125+6=-125

125+7=-124

125+8=-123

125+9=-122

125+10=-121

3. What will be output if you will compile and execute the following c code?

struct marks{

  int p:3;

  int c:3;

  int m:2;

};

void main(){

  struct marks s={2,-6,5};

  printf("%d %d %d",s.p,s.c,s.m);

}

(a) 2 -6 5

(b) 2 -6 1

(c) 2 2 1

(d) Compiler error

(e) None of these

Answer: (c)

Explanation:

Binary value of 2: 00000010 (Select three two bit)

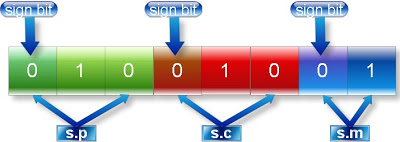
Binary value of 6: 00000110

Binary value of -6: 11111001+1=11111010

(Select last three bit)

Binary value of 5: 00000101 (Select last two bit)

Complete memory representation:



4. What will be output if you will compile and execute the following c code?

void main(){

   int a=25;

   clrscr();

   printf("%o %x",a,a);

   getch();

}

(a) 25 25

(b) 025 0x25

(c) 12 42

(d) 31 19

(e) None of these

Answer: (d)

Explanation:

%o is used to print the number in octal number format.

%x is used to print the number in hexadecimal number format.

Note: In c octal number starts with 0 and hexadecimal number starts with 0x

5.

Study the following 'C' program

void main()

{

static a,b;

while(a > b++)

}

What will be the value of a and b?

A.a = 0, b = 1 B. a = 0, b = 0

C.a = 1, b = 1 D. None of these

Ans : A

6. Consider the following program fragment

switch(input)

{

case '1':

printf("One");

case '2':

printf("Two");

case '3':

printf(""Three");

default:

Printf("Default");

break;

}

What will be printed when input is 2?

A. Two Three Default B. Two

C. Two Default D. Two Two Default

Ans : A

7.What will be printed when this program is executed?

int f(int x)

{

if(x <= 4)

return x;

return f(--x);

}

void main()

{

printf("%d ", f(7));

}

A. 4 5 6 7

B. 1 2 3 4

C. 4

D. Syntax error

E. Runtime error

Answer: Option C

Solution:

In this recursive function call the function will return to main caller when the value of x is 4. Hence the output.

Bottom of Form

8.The recursive functions are executed in a ...........

A. Parallel order

B. First In First Out order

C. Last In First Out order

D. Iterative order

E. Random order

Answer: Option C

Solution:

Because for each function call an entry is created in stack frame( known as Active Record Instance), and are executed in LIFO manner.

9.

char\* myfunc(char \*ptr)

{

ptr+=3;

return(ptr);

}

void main()

{

char \*x, \*y;

x = "EXAMVEDA";

y = myfunc(x);

printf("y=%s", y);

}

What will be printed when the sample code above is executed?

A. y=EXAMVEDA

B. y=MVEDA

C. y=VEDA

D. y=EDA

E. y=AMVEDA

Answer: Option B

10.

## What will be the output?

main()

{

char \*p;

p = "Hello";

printf("%cn",\*&\*p);

}

A. Hello

B. H

C. Some address will be printed

D. None of these.

Answer: Option B

Solution:

\* is a dereference operator & is a reference operator. They can be applied any number of times provided it is meaningful. Here p points to the first character in the string "Hello". \*p dereferences it and so its value is H. Again & references it to an address and \* dereferences it to the value H.

## 11. Find the output of the following program.

void main()

{

int array[10];

int \*i = &array[2], \*j = &array[5];

int diff = j-i;

printf("%d", diff);

}

A. 3

B. 6

C. Garbage value

D. Error

Answer: Option A

Solution:

When subtracting pointers you get the number of elements between those addresses, not the number of bytes.

12.

main()

{

char s[ ]="man";

int i ;

for(i=0;s[ i];i++)

printf("\n%c%c%c%c",s[i],\*(s+i),\*(i+s),i[s]);

}

A) mmmm

aaaa

nnnn

B) mmm

aaa

nnn

C)nnn

aaa

mmmm

D)aaa

mmm

nnnnn

Answer:A

Explanation:

s[i], \*(i+s), \*(s+i), i[s] are all different ways of

expressing the same idea. Generally array name is the

base address for that array. Here s is the base address.

i is the index number/displacement from the base

address. So, indirecting it with \* is same as s[i].i[s] may

be surprising. But in the case of C it is same as s[i].

13.main()

{

extern int i;

i=20;

printf("%d",i);

}

A)20

B)Linkr error

C) Garbage Value

D)Compiler Error

Answer:B

Linker Error : Undefined symbol '\_i'

Explanation:

extern storage class in the following declaration,

extern int i;

specifies to the compiler that the memory for i

is allocated in some other program and that address will

be given to the current program at the time of linking.

But linker finds that no other variable of name

i is available in any other program with memory space

allocated for it. Hence a linker error has occurred .

14.Predict the output or error(s) for the following:

main()

{

int i=-1,j=-1,k=0,l=2,m;

m=i++&&j++&&k++||l++;

printf("%d %d %d %d %d",i,j,k,l,m);

}

A)00131

B)13100

C)13001

D)13010

Answer:A

Explanation :

Logical operations always give a result of 1 or 0 . And

also the logical AND (&&) operator has higher priority

over the logical OR (||) operator. So the expression

‘i++&& j++ && k++’ is executed first. The result of this

expression is 0(-1 && -1 && 0 = 0). Now the

expression is 0 || 2 which evaluates to 1 (because OR

operator always gives 1 except for ‘0 || 0’ combination-

for which it gives 0). So the value of m is 1. The values

of other variables are also incremented by 1.

15. What is the output of following C code?

main()

{

struct emp

{

char name[20];

int age;

float sal;

};

struct emp e ={"Tiger"}

printf("%d%d%f",e.age,e.sal);

}

[A] Error

[B] Garbage Collection

[C] 0 0.000000

[D] 1 0.000000

Answer: C. 0 0.000000

16. Which of the following comment about the usage of structures in true?

[A] Storage class can be assigned to individual member

[B] Individual members can be initialized within a structure type declaration

[C] The scope of the member name is confined to the particular structure, within which it is defined

[D] None of above

Answer: C. The scope of the member name is confined to the particular structure, within which it is defined

Explanation: Structure is user defined data type which is used to store heterogeneous data under unique name.

17.Which of the following comment about Union is false?

[A] Union is a structure whose members share same memory area

[B] The compiler will keep track of what type of information is currently stored

[C] Only one of the members of union can be assigned a value at particular time

[D] Size allocated for Union is the size of its member needing the maximum storage

Answer: B. The compiler will keep track of what type of information is currently stored

Explanation: Union is similar to structure the only difference is the way the memory allocated to structure and union

18.What will be the output of teh following program ?

#include

main()

{

structxx

{

intx=3;

char name[]="hello";

};

struct xx \*s;

printf("%d",s->x);

printf("%s",s->name);

}

A)hello

B)compilation ERROR

C)hello

hello

D)No output

Answer:B

Explanation:

You should not initialize variables in declaration

19.

what will be the output for the following ?

main()

{

printf("\nab");

printf("\bsi");

printf("\rha");

}

Answer:

hai

Explanation:

\n - newline

\b - backspace

\r - linefeed

20. main()

{

int i=400,j=300;

printf("%d..%d");

}

Answer:

400..300

Explanation:

printf takes the values of the first two assignments of the program. Any number of

printf's may be given. All of them take only the first two values. If more number of

assignments given in the program, then printf will take garbage values.