Practice Sheet

**1. void main()**

**{**

**int a[5]={10,20,30,40,50};**

**int \*ptr;**

**ptr=a+1;**

**printf("%d %d %d",++\*ptr,\*ptr++,\*++ptr);**

**}**

**2. void main()**

**{**

**int a[5]={6,16,26};**

**++\*a;**

**++\*(a+1);**

**++a[1];**

**printf("\n%d %d",a[0],a[1]);**

**}**

**3. void main()**

**{**

**int a[]={10,20,30,40},\*p;**

**p=a+1;**

**printf("%d %d %d %d %d\n",-1[p],p[-1],0[p],p[1],\*(p+2));**

**}**

**4. void main()**

**{**

**char s1[8],s2[8];**

**scanf("%s%s",s1,s2);**

**printf("char1=%c,char2=%c",s1[52-'0'],s2[\*s2-0[s2]]+4);**

**}**

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

**void main()**

**{**

**char \*n;**

**scanf("%s", n); //Enter: 10**

**printf("%s",n);**

**}**

**6. void main(){**

**int huge\*p=(int huge\*)0XC0563331;**

**int huge\*q=(int huge\*)0xC2551341;**

**\*p=200;**

**printf("%d",\*q);**

**}**

**7. int \*p, x;**

**int a[5]={100,200,300,400,500};**

**int \*p2;**

**void main()**

**{**

**p=NULL;**

**x=500;**

**p=&x;**

**printf("1) %d %d \n",x,\*p,p,&x,&p);**

**p2=a;**

**\*(p2+1)=\*p;**

**\*p= \*p2 + \*(p2+2);**

**printf("2) %d %d \n",x,\*p,\*p2);**

**}**

**8. What is the output of the following program, assuming that**

**the address of x is 003674D0, the address of a is 00367038?**

**int \*p, x,y;**

**int a[5]={ 100,200,300,400,500};**

**int \*p2;**

**void main()**

**{**

**p=NULL;**

**x=10;**

**p=&x;**

**printf("1) %d %d %p %p %p \n",x,\*p,p,&x,&p);**

**p2=&x;**

**printf("2) %d %d %p %p \n",x,\*p2,p2,&x);**

**p2=a;**

**printf("3) %d %d %p %p \n",a[0],\*p2,p2,a);**

**p2=&a[2];**

**printf("4) %d %p %p \n",\*p2,p2,a);**

**p2++;**

**printf("5) %d %p \n",\*p2,p2);**

**p=a;**

**y=\*(p+2);**

**printf("6) %d %d \n",y,a[2]);**

**printf(" loop using pointer \n");**

**for ( int \* p3=a; p3 <= &a[4]; p3++)**

**printf("loop: %d %p \n",\*p3,p3);**

**}**

**9. void main(){**

**char c=256;**

**char \*ptr="Leon";**

**if(c==0)**

**while(!c)**

**if(\*ptr++)**

**printf("%+u",c);**

**else**

**break;**

**}**

**10.int a[50];**

**int \*pa;**

**pa=a;**

**To access the 6th element of the array which of the following**

**is incorrect?**

**(a) \*(a+5) (b) a[5] (c) pa[5] (d) \*(\*pa + 5}**

**ans:d**

**11.What will be output of following program?**

**void main(){**

**int huge \*a =(int huge \*)0x59990005;**

**int huge \*b =(int huge \*)0x59980015;**

**if(a == b)**

**printf("power of pointer");**

**else**

**printf("power of c");**

**}**

**12.void main(){**

**int x=10;**

**int far \*ptr;**

**ptr=&x;**

**printf("%d",sizeof ptr);**

**}**

**13.void main(){**

**int far \*near\*ptr;**

**printf("%d %d",sizeof(ptr) ,sizeof(\*ptr));**

**}**

**14.void main(){**

**int far \*p,far \*q;**

**printf("%d %d",sizeof(p) ,sizeof(q));**

**}**

**15.void main(){**

**int x=100;**

**int far \*ptr;**

**ptr=&x;**

**printf("%Fp",ptr);**

**}**

**16.#include "dos.h"**

**void main(){**

**int i=25;**

**int far\*ptr=&i;**

**printf("%X %X",FP\_SEG(ptr),FP\_OFF(ptr));**

**}**

**17.#include "dos.h"**

**void main(){**

**int i=25;**

**int far\*ptr=&i;**

**unsigned int s,o;**

**s=FP\_SEG(ptr);**

**o=FP\_OFF(ptr);**

**printf("%Fp",MK\_FP(s,o));**

**}**

**18.void main(){**

**int i;**

**char far \*ptr=(char \*)0xB800FFFA;**

**for(i=0;i<=10;i++){**

**printf("%Fp \n",ptr);**

**ptr++;**

**}**

**}**

**19.void main(){**

**int far \*p=(int \*)0X70230000;**

**int far \*q=(int \*)0XB0210000;**

**if(p==q)**

**printf(" equal");**

**else**

**printf(" not equal");**

**}**

**20.void main(){**

**int far \*p=(int \*)0X70230000;**

**int far \*q=(int \*)0XB0210000;**

**int near \*x,near\*y;**

**x=(int near \*)p;**

**y=(int near \*)q;**

**if(x==y)**

**printf(" equal");**

**else**

**printf(" not equal");**

**}**

**21.How to read following pointer?**

**char (\* ptr)[3]**

**22.How to read following pointer?**

**float (\* ptr)(int)**

**23.void (\*ptr)(int (\*)[2],int (\*) void))**

**24.How to read following pointer?**

**int ( \* ( \* ptr ) [ 5 ] ) ( )**

**25.char far \* display(char far\*);**

**void main(){**

**char far\* string=" ";**

**char far \*(\*ptr)(char far \*);**

**ptr=&display;**

**string=(\*ptr)(string);**

**printf("%s",string);**

**}**

**char far \*display(char far \* str){**

**char far \* temp=str;**

**temp=temp+13;**

**\*temp='\0';**

**return str;**

**}**

**26.char \* call(int \*,float \*);**

**void main(){**

**char \*string;**

**int a=2;**

**float b=2.0l;**

**char \*(\*ptr)(int\*,float \*);**

**ptr=&call;**

**string=(\*ptr)(&a,&b);**

**printf("%s",string);**

**}**

**char \*call(int \*i,float \*j){**

**static char \*str="Java Professional";**

**str=str+\*i+(int)(\*j);**

**return str;**

**}**

**27.int find(char);**

**int(\*function())(char);**

**void main(){**

**int x;**

**int(\*ptr)(char);**

**ptr=function();**

**x=(\*ptr)('A');**

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

**}**

**int find(char c){**

**return c;**

**}**

**int(\*function())(char){**

**return find;**

**}**

**28.int \* function();**

**void main(){**

**auto int \*x;**

**int \*(\*ptr)();**

**ptr=&function;**

**x=(\*ptr)();**

**printf("%d",\*x);**

**}**

**int \*function(){**

**static int a=10;**

**return &a;**

**}**

**29.int \*call();**

**void main(){**

**int \*ptr;**

**ptr=call();**

**fflush(stdin);**

**printf("%d",\*ptr);**

**}**

**int \* call(){**

**int x=25;**

**++x;**

**return &x;**

**}**

**30.void main(){**

**static double \*p,\*q,\*r,\*s,t=5.0;**

**double \*\*arr[]={&p,&q,&r,&s};**

**int i;**

**\*p=\*q=\*r=\*s=t;**

**for(i=0;i<4;i++)**

**printf("%.0f  ",\*\*arr[i]);**

**}**

**31.void main(){**

**printf("%d %d",2["ABCDEFGH"],"ABCDEFGH"[5]);**

**}**

**32.void main(){**

**int a[5];**

**int (\*b)[5];**

**printf("%d %d %d",sizeof(a),sizeof(\*b),sizeof(b));**

**}**

**33.void main(){**

**void (\*p)();**

**int (\*q)();**

**int (\*r)();**

**p = clrscr;**

**q = getch;**

**r = puts;**

**(\*p)();**

**(\*r)("abc");**

**(\*q)();**

**}**

**34.void main(){**

**char far \*p =(char far \*)0x55550005;**

**char far \*q =(char far \*)0x53332225;**

**\*p = 80;**

**(\*p)++;**

**printf("%d",\*q);**

**}**

**35.void main(){**

**char far \*p,\*q;**

**printf("%d %d",sizeof(p),sizeof(q));**

**}**

**36.void main(){**

**int a = 10;**

**void \*p = &a;**

**int \*ptr = p;**

**printf("%u",\*ptr);**

**}**

**37.void main(){**

**int a,b,c,d;**

**char \*p = ( char \*)0;**

**int \*q = ( int \*q)0;**

**float \*r = ( float \*)0;**

**double \*s = 0;**

**a = (int)(p+1);**

**b = (int)(q+1);**

**c = (int)(r+1);**

**d = (int)(s+1);**

**printf("%d %d %d %d",a,b,c,d);**

**}**

**38.void main(){**

**int a = 5,b = 10,c;**

**int \*p = &a,\*q = &b;**

**c = p - q;**

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

**}**

**39.unsigned long int (\* avg())[3]{**

**static unsigned long int arr[3] = {1,2,3};**

**return &arr;**

**}**

**void main(){**

**unsigned long int (\*ptr)[3];**

**ptr = avg();**

**printf("%d" , \*(\*ptr+2));**

**}**

**(A) 1 (B) 2 (C) 3 (D) Compilation error (E) None of above**

**40.void main(){**

**int i = 3;**

**int \*j;**

**int \*\*k;**

**j = &i;**

**k = &j;**

**printf("%u %u %u",i,j,k);**

**}**

**41.void main(){**

**int a = 320;**

**char \*ptr;**

**ptr =(char \*)&a;**

**printf("%d ",\*ptr);**

**}**

**42.void main(){**

**printf("%c",\*"abcde");**

**}**

**43.void main(){**

**static double \*p,\*q,\*r,\*s,t=5.0;**

**double \*\*arr[]={&p,&q,&r,&s};**

**int i;**

**\*p=\*q=\*r=\*s=t;**

**for(i=0;i<4;i++)**

**printf("%.0f  ",\*\*arr[i]);**

**}**

**44.void main()**

**{**

**char \*p = NULL;**

**char \*q = 0;**

**if(p)**

**printf(" p ");**

**else**

**printf("nullp");**

**if(q)**

**printf("q\n");**

**else**

**printf(" nullq\n");**

**}**

**45.int \*f();**

**void main()**

**{**

**int \*p = f();**

**printf("%d\n", \*p);**

**}**

**int \*f()**

**{**

**int j = 10;**

**return &j;**

**}**

**46.Comment on the following pointer declaration?**

**int\* ptr, p;**

**a) ptr is a pointer to integer, p is not**

**b) ptr and p, both are pointers to integer**

**c) ptr is a pointer to integer, p may or may not be**

**d) ptr and p both are not pointers to integer**

**47. What is the output of this C code?**

**void main()**

**{**

**int \*ptr, a = 10;**

**ptr = &a;**

**\*ptr += 1;**

**printf("%d,%d/n", \*ptr, a);**

**}**

**48. Comment on the following?**

**const int \*ptr;**

**a) You cannot change the value pointed by ptr**

**b) You cannot change the pointer ptr itself**

**c) Both (   a) and (b)**

**d) You can change the pointer as well as the value pointed by**

**it**

**49. Which is an indirection operator among the following?**

**a) &**

**b) \***

**c) ->**

**d) .**

**50. Which of the following does not initialize ptr to null**

**(assuming variable declaration of a as int    a=0;?**

**a) int \*ptr = &a;**

**b) int \*ptr = &a – &a;**

**c) int \*ptr = a – a;**

**d) All of the mentioned**

**51.int x = 0;**

**void main()**

**{**

**int \*ptr = &x;**

**printf("%p\n", ptr);**

**x++;**

**printf("%p\n ", ptr);**

**}**

**a) Same address**

**b) Different address**

**c) Compile time error**

**d) Varies**

**52.What is the output of this C code?**

**int x = 0;**

**void main()**

**{**

**int \*const ptr = &x;**

**printf("%p\n", ptr);**

**ptr++;**

**printf("%p\n ", ptr);**

**}**

**53.What is the output of this C code?**

**void main()**

**{**

**int x = 0;**

**int \*ptr = &5;**

**printf("%p\n", ptr);**

**}**

**a) 5**

**b) Address of 5**

**c) Nothing**

**d) Compile time error**

**54.What is the output of this C code?**

**void main()**

**{**

**int x = 0;**

**int \*ptr = &x;**

**printf("%d\n", \*ptr);**

**}**

**55.What is the output of this C code?**

**void foo(int\*);**

**void main()**

**{**

**int i = 10;**

**foo((&i)++);**

**}**

**void foo(int \*p)**

**{**

**printf("%d\n", \*p);**

**}**

**56.What is the output of this C code?**

**void foo(int\*);**

**void main()**

**{**

**int i = 10, \*p = &i;**

**foo(p++);**

**}**

**void foo(int \*p)**

**{**

**printf("%d\n", \*p);**

**}**

**a) 10**

**b) Some garbage value**

**c) Compile time error**

**d) Segmentation fault**

**57.What is the output of this C code?**

**void foo(float \*);**

**void main()**

**{**

**int i = 10, \*p = &i;**

**foo(&i);**

**}**

**void foo(float \*p)**

**{**

**printf("%f\n", \*p);**

**}**

**58.What is the output of this C code?**

**void foo(int\*);**

**void main()**

**{**

**int i = 97, \*p = &i;**

**foo(&i);**

**printf("%d ", \*p);**

**}**

**void foo(int \*p)**

**{**

**int j = 2;**

**p = &j;**

**printf("%d ", \*p);**

**}**

**a) 2 97**

**b) 2 2**

**c) Compile time error**

**d) Segmentation fault/code crash**

**59.What is the output of this C code?**

**void main()**

**{**

**int i = 97, \*p = &i;**

**foo(&p);**

**printf("%d ", \*p);**

**}**

**void foo(int \*\*p)**

**{**

**int j = 2;**

**\*p = &j;**

**printf("%d ", \*\*p);**

**}**

**60.What is the output of this C code?**

**void main()**

**{**

**int i = 11;**

**int \*p = &i;**

**foo(&p);**

**printf("%d ", \*p);**

**}**

**void foo(int \*const \*p)**

**{**

**int j = 10;**

**\*p = &j;**

**printf("%d ", \*\*p);**

**}**

**61.What is the output of this C code?**

**void main()**

**{**

**int i = 10;**

**int \*p = &i;**

**foo(&p);**

**printf("%d ", \*p);**

**printf("%d ", \*p);**

**}**

**void foo(int \*\*const p)**

**{**

**int j = 11;**

**\*p = &j;**

**printf("%d ", \*\*p);**

**}**

**62.What is the output of the code below?**

**void main()**

**{**

**int i = 10;**

**int \*const p = &i;**

**foo(&p);**

**printf("%d\n", \*p);**

**}**

**void foo(int \*\*p)**

**{**

**int j = 11;**

**\*p = &j;**

**printf("%d\n", \*\*p);**

**}**

**63.Which of the following are correct syntaxes to send an array**

**as a parameter to function:**

**a) func(&array);**

**b) func(array);**

**c) func(\*array);**

**d) func(array[size]);**

**64.Which of the following can never be sent by call-by-value?**

**a) Variable**

**b) Array**

**c) Structures**

**d) Both (   b) and (c) s**

**65.What is the output of this C code?**

**void m(int \*p, int \*q)**

**{**

**int temp = \*p; \*p = \*q; \*q = temp;**

**}**

**void main()**

**{**

**int a = 6, b = 5;**

**m(&a, &b);**

**printf("%d %d\n", a, b);**

**}**

**66 void foo( int[] );**

**void main()**

**{**

**int ary[4] = {1, 2, 3, 4};**

**foo(ary);**

**printf("%d ", ary[0]);**

**}**

**void foo(int p[4])**

**{**

**int i = 10;**

**p = &i;**

**printf("%d ", p[0]);**

**}**

**67.What is the output of the code given below?**

**void main()**

**{**

**int ary[4] = {1, 2, 3, 4};**

**int \*p = ary + 3;**

**printf("%d\n", p[-2]);**

**}**

**68.What is the output of the code given below?**

**void main()**

**{**

**int ary[4] = {1, 2, 3, 4};**

**int \*p = ary + 3;**

**printf("%d %d\n", p[-2], ary[\*p]);**

**}**

**69.What is the output of this C code?**

**void main()**

**{**

**int ary[4] = {1, 2, 3, 4};**

**printf("%d\n", \*ary);**

**}**

**70.What is the output of this C code?**

**void main()**

**{**

**static int ary[4] = {1, 2, 3, 4};**

**a[1]=10;**

**printf("%d",a[1]);**

**}**

**71.What is the output of this C code?**

**void main()**

**{**

**static int ary[4] = {1, 2, 3, 4};**

**\*a=10;**

**printf("%d",a[1]);**

**}**

**72.What is the output of this C code?**

**void main()**

**{**

**static int a[4] = {1, 2, 3, 4};**

**\*(a+1)=10;**

**printf("%d",a[1]);**

**}**

**73.Different ways to initialize an array with all elements as**

**zero are**

**a) int array[5] = {};**

**b) int array[5] = {0};**

**c) int a = 0, b = 0, c = 0;**

**int array[5] = {a, b, c};**

**d) All of the mentioned**

**74. void main() {**

**char str[]="abcdef";**

**++str;**

**++\*str;**

**puts(str);**

**}**

**75. void main() {**

**int a[5]={1,2,3};**

**++\*a;**

**\*a++;**

**printf("\n%d %d",a[0],a[1]);**

**}**

**76. void main() {**

**int a[5]={1,2,3,4,5};**

**int \*ptr;**

**ptr=a+1;**

**printf("%d %d %d",++\*ptr,\*ptr++,\*++ptr);**

**}**

**77. what is the use of pointers.**

**a. accessing array elements**

**b. passing argument to functions by reference.**

**c. passing arrays and strings to functions**

**d. creating data structure such as linked lists, trees,**

**graphs**

**e. obtaining memory from the system dynamicaly**

**f. all of the given.**

**78. void main(){**

**int x=10;**

**printf("%d %d",x,\*&x);**

**}**

**79.void main(){**

**int x=10,\*px=&x;**

**printf("%d %d",x,\*px);**

**}**

**80. void main(){**

**int x=10,y=20;**

**int \*p=&x;**

**\*p=1;**

**p=&y;**

**\*p=2;**

**printf("%d %d %d",x,y,\*p);**

**}**

**81.void main(){**

**int a1=20,a2=10;**

**char b1=20,b2=10;**

**float c1=30,c2=30;**

**printf("%d %d %d",&a1-&a2,&b1-&b2,&c1-&c2);**

**}**

**82.void main(){**

**int x=1,\*p=&y;**

**x++;**

**(\*p)++;**

**printf("%d %d",x,\*p);**

**}**

**83.void fun(int a,int\* b){**

**a=a+\*b;**

**\*b=a-\*b;**

**a=a-\*b;**

**}**

**void main(){**

**int x=10,y=20;**

**fun(x,&y);**

**printf("%d %d",x,y);**

**}**

**84.void main(){**

**int \*x;**

**char \*y;**

**void \*z;**

**int a=sizeof(x)+sizeof(y)+sizeof(z);**

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

**}**

**85. void main(){**

**printf("%d %d ",sizeof(int\*),sizeof(float\*));**

**printf("%d %d",sizeof(char\*),sizeof(void\*));**

**}**

**86.void main(){**

**int \*x;**

**char \*y;**

**float \*z;**

**int a=sizeof(\*x)+sizeof(\*y)+sizeof(\*z);**

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

**}**

**87.void main(){**

**int \*iptr,var1,var2;**

**iptr=&var1;**

**\*iptr=25;**

**\*iptr+=10;**

**printf("%d ",var1);**

**var2=\*iptr;**

**printf("%d ",var2);**

**iptr=&var2;**

**\*iptr+=20;**

**printf("%d %d",var1,var2);**

**}**

**88. which is false**

**a. a pointer to an int accesses 2 consecutive bytes of**

**memory**

**b. a pointer to an char accesses 1 byte of memory**

**c. a pointer to an float access 4 consecutive bytes of**

**memory**

**d. a memory bytes pointed by void pointer is system**

**dependent**

**89. the arithmetic operators which applicable for pointers are**

**a. ++,--, unary(-), unary(+)**

**b. ++,--, binary(-), binary(+)**

**c. ++,--, unary(-), unary(+),\*,/**

**d. ++,--,<<,>>, binary(-), binary(+)**

**90. select the valid expression for the given**

**declaration(multiple).**

**int a,b,\*p,\*q;**

**a. p=-q;**

**b. p<<=1;**

**c. p=p-b;**

**d. p=p-q;**

**91. select the valid expression for the given**

**declaration(multiple).**

**int a,b,\*p,\*q;**

**a. p=(int\*)(p-q);**

**b. p=p-q-a;**

**c. p=p\*a;**

**d. p=p+a;**

**e. p=p+q;**

**92. select the valid expression for the given**

**declaration(multiple).**

**int a,b,\*p,\*q;**

**a. p=p+q+a;**

**b. p=p\*q;**

**c. p=(int\*)(p-q)-a;**

**d. p=p/q;**

**e. p=p/b;**

**f. p=a/p;**

**93. void main(){**

**char near \*p1;**

**char far \*p2;**

**char huge \*p3;**

**printf("%d %d %d ",sizeof(p1)+sizeof(p2)+sizeof(p3));**

**printf("%d %d %d",sizeof(\*p1)+sizeof(\*p2)+sizeof(\*p3));**

**}**

**94.void main(){**

**int x=1,\*p=&x,\*\*pp=&p;**

**x++;**

**++\*p;**

**++\*\*pp;**

**printf("%d %d %d",x,\*p,\*\*pp);**

**}**

**95.void main(){**

**int a[3]={1,2,3};**

**int\* p=a+1;**

**printf("%d %d %d ",a[1],p[1],p[-1]);**

**printf("%d %d %d %d ",-a[1],1[a],-p[1],-1[p]);**

**printf("%d %d",\*(p+1),\*(a+1));**

**}**

**96.void fun(int a[5],int b[],int \*c){**

**printf("%d %d %d",sizeof(a),sizeof(b),sizeof(c));**

**}**

**void main(){**

**int a[5];**

**fun(a,a,a);**

**}**

**97.void fun(char a[5],char b[],char \*c){**

**printf("%d %d %d",sizeof(a),sizeof(b),sizeof(c));**

**}**

**void main(){**

**char a[]="abcd";**

**fun(a,a,a);**

**}**

**98.int\* f1(){**

**int x=10;**

**return &x;**

**}**

**int f2(){**

**int x=1;**

**return x+1;**

**}**

**void main(){**

**int\* x,y;**

**x=f1();**

**y=f2();**

**printf("%d %d",\*x,y);**

**}**

**99.int\* fun(int\* p){**

**++\*p;**

**return p;**

**}**

**void main(){**

**static int x;**

**\*fun(&x)=x+1;**

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

**}**

**100.int\* fun(int\* p){**

**++\*p;**

**return p;**

**}**

**void main(){**

**static int x;**

**\*fun(&x)+=1;**

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

**}**

**101.int[] f1(){**

**int a[3]={1,2};**

**return a;**

**}**

**void main(){**

**printf("%d",sizeof(f1()));**

**}**

**102.void main(){**

**int x = 4;**

**int \*p = &x;**

**int \*k = p++;**

**int r = p - k;**

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

**}**

**103.void main()**

**{**

**const int i = 10;**

**int \*ptr = &i;**

**\*ptr = 20;**

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

**}**

**104.void main(){**

**int\* p1;**

**int p2[5];**

**int (\*p3)[2];**

**printf("%d",sizeof(p1),sizeof(p2),sizeof(p3));**

**}**

**105.void main(){**

**int a=262,b;**

**char \*ptr;**

**ptr=&a;**

**b=\*ptr;**

**printf("\n%d %d",a,b);**

**}**

**106.void main() {**

**int arr[5]={9,19,29,39,49};**

**int \*ptr;**

**ptr=arr+2;**

**++\*ptr;**

**printf("\n%d %d %d",++\*ptr,\*ptr++,\*++ptr);**

**}**

**107.void abc(int \*ptr){**

**++\*ptr;**

**if(\*ptr<=10)**

**abc(++ptr);**

**++\*ptr;**

**}**

**void main() {**

**int arr[5]={3,6,9,11,13};**

**int i;**

**abc(arr);**

**for(i=0;i<5;i++)**

**printf("%d ",arr[i]);**

**}**

**108.int x=10;**

**int\* fun(){**

**return &x;**

**}**

**void main(){**

**\*fun()=1;**

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

**}**

**109.void main(){**

**int x=1,y=2;**

**int\* p=&y;**

**\*p=10;**

**p++;**

**\*p=20;**

**printf("%d %d",x,y);**

**}**

**110.void main() {**

**int arr[][3]={**

**{1,2,3},**

**{4,5,6},**

**{7,8,9}**

**};**

**int \*ptr[3];**

**ptr[0]=arr; ptr[1]=arr+1;**

**ptr[2]=arr+2; ++ptr[0];**

**--\*ptr[0];  ++ptr[1];**

**++\*ptr[1];  ++ptr[2];**

**\*ptr[2]=30;**

**printf("\n%d %d %d",\*ptr[0],\*ptr[1],\*ptr[2]);**

**printf("\n%d %d %d",\*\*ptr,\*\*(ptr+1),\*\*(ptr+2));**

**}**

**111.void main() {**

**int arr[5]={5,15,25,35,45};**

**int near\*ptr=NULL;**

**ptr=&arr[0];**

**++ptr;  ++\*ptr;**

**--ptr;  --\*ptr;**

**printf("\n%d %d",arr[0],arr[1]);**

**}**

**112.void main()  {**

**int arr[5]={6,16,26,36,46};**

**int near\*ptr;**

**ptr=arr+2;**

**++ptr;**

**++\*ptr;**

**--ptr;**

**--\*ptr;**

**printf("\n%d %d",arr[2],arr[3]);**

**}**

**113.void main(){**

**int arr[5]={8,18,28,38,48};**

**int near\*ptr=arr+2;**

**--ptr;**

**++\*ptr;**

**++ptr;**

**--\*ptr;**

**printf("\n%d %d %d",ptr[0],ptr[1],ptr[-1]);**

**}**

**114.void main() {**

**int arr[5]={4,14,24,34,44};**

**int \*ptr;**

**ptr=arr+1;**

**++ptr;**

**++\*ptr;**

**--ptr;**

**--\*ptr;**

**printf("\n%d %d %d",\*(arr+0),\*(arr+1),\*(arr+2));**

**}**

**115.void main(){**

**int \*a=(int\*)100,x=10;**

**\*a=1;**

**a=&x;**

**printf("%d %d %d",\*a,x,\*((int\*)100));**

**}**

**116.what is the output**

**void main(){**

**int a[3][4]={{1,2,3,4},{5,6,7,8},{9,10,11,12}};**

**int \*p=&a[1][6];**

**printf("%d",\*p);**

**}**

**117.What will be output of the following c program?**

**void main(){**

**int a[]={6,7,8,9},i;**

**compute(a);**

**for(i=3;i>=0;i--)**

**printf(“%d”,a[i]);**

**}**

**compute(int \*p){**

**int i;**

**for(i=0;i<4;i++){**

**\*p=\*p-1;**

**p++;**

**}**

**}**

**118.The array a[j][k] is equivalent to**

**(A) ((base type\*)a+(j\*row length)+k)**

**(B) \*((base type\*)a+(j\*row length)+k)**

**(C) \*((base type)a+(j\*row length)+k)**

**(D) \*((base type)a+(j\*row length))**

**(E) None of these**

**119.void main()**

**{**

**int x=10; //address of x is 100**

**int\* p=&x;**

**printf("%d",x\*p);**

**}**

**120.void main()**

**{**

**int x=10;**

**int\* p=&x;**

**printf("%d",x/\*p);**

**}**

**121. void main()**

**{**

**int a=1,b=2,c=3,x=10,y=20;**

**scanf("%d%d%d",&c+1,&c-1,&c+2);**

**//Entered values: 11 12 13**

**printf("%d %d %d %d %d",a,b,c,x,y);**

**}**

**122.void main()**

**{**

**int a[]={1,2};**

**int\* p=a;**

**++\*p++;**

**printf("%d %d %d",\*p,a[0],a[1]);**

**}**

**123.int\* abc(int a)**

**{**

**static int s;**

**s=++a;**

**return &s;**

**}**

**void main()**

**{**

**int \*p;**

**int x=100;**

**int r;**

**int \*(\*ptr)(int);**

**ptr=&abc;**

**p=ptr(x);**

**r=\*ptr(x);**

**printf("\n%d %d",r,\*p);**

**}**

**124:void main()**

**{**

**int arr[][3]={**

**{1,2,3},**

**{4,5,6},**

**{7,8,9}**

**};**

**int \*ptr[3];**

**ptr[0]=arr;**

**ptr[1]=arr+1;**

**ptr[2]=arr+2;**

**++ptr[0];**

**--\*ptr[0];**

**++ptr[1];**

**++\*ptr[1];**

**++ptr[2];**

**\*ptr[2]=30;**

**printf("\n%d %d %d",\*ptr[0],\*ptr[1],\*ptr[2]);**

**printf("\n%d %d %d",\*\*ptr,\*\*(ptr+1),\*\*(ptr+2));**

**}**

**125:void main()**

**{**

**int arr[3][3]={**

**{10,20,30},**

**{40,50,60},**

**{70,80,90}**

**};**

**int \*ptr[3];**

**int \*\*pptr;**

**ptr[0]=&arr[0][0];**

**ptr[1]=&arr[1][0];**

**ptr[2]=&arr[2][0];**

**pptr=ptr+1;**

**++pptr;**

**++\*pptr;**

**++\*\*pptr;**

**++\*ptr[2];**

**--pptr;**

**++\*pptr;**

**--\*\*pptr;**

**--\*ptr[1];**

**--pptr;**

**++\*pptr;**

**--\*\*pptr;**

**++\*ptr[0];**

**printf("\n%d %d %d",\*ptr[0],\*ptr[1],\*ptr[2]);**

**printf("\n%d %d %d",\*\*pptr,\*\*(pptr+1),\*\*(pptr+2));**

**}**

**127:void main()**

**{**

**int arr[3][4]={**

**{10,20,30,40},**

**{11,21,31,41},**

**{12,22,32,42}**

**};**

**int \*ptr[3];**

**int \*\*pptr;**

**int i,r,c;**

**ptr[0]=arr;**

**ptr[1]=arr+1;**

**ptr[2]=arr+2;**

**pptr=ptr;**

**for(i=3;i>=1;i--)**

**{**

**\*pptr+=i;**

**\*\*pptr+=i;**

**++pptr;**

**}**

**--pptr;**

**printf("\n%d\n",\*\*pptr);**

**for(i=0;i<3;i++)**

**printf("%d ",\*ptr[i]);**

**printf("\narr List is: ");**

**for(r=0;r<3;r++)**

**{**

**printf("\n");**

**for(c=0;c<4;c++)**

**{**

**printf("%d\t",arr[r][c]);**

**}**

**}**

**}**

**128.int\* abc()**

**{**

**static int s=10;**

**++s;**

**return &s;**

**}**

**void main()**

**{**

**int \*p;**

**int (\*ptr)();**

**ptr=&abc;**

**p=ptr();**

**printf("\n%d",\*p);**

**\*p=20;**

**printf("\n%d",\*p);**

**}**

**129.int\*abc(int a)**

**{**

**static int s;**

**s=++a;**

**return &s;**

**}**

**void main()**

**{**

**int \*p;**

**int x=100;**

**int r;**

**int \*(\*ptr)(int);**

**ptr=&abc;**

**p=ptr(x);**

**r=\*ptr(x);**

**printf("\n%d %d",r,\*p);**

**}**

**130.int\*funptr(int a)**

**{**

**static int s;**

**s=++a;;**

**return &s;**

**}**

**void main()**

**{**

**int \*p;**

**int \*(\*ptr)(int);**

**ptr=&funptr;**

**p=ptr(10);**

**printf("\nValue=%d",\*p);**

**}**

**131.int funptr()**

**{**

**static int s=10;**

**++s;**

**return s;**

**}**

**void main()**

**{**

**int r;**

**int (\*ptr)();**

**ptr=&funptr;**

**r=ptr();**

**printf("\nValue=%d",r);**

**}**

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

**void xyz(int \*ptr)**

**{**

**++\*ptr;**

**if(\*ptr<=5)**

**xyz(ptr+1);**

**++\*ptr;**

**}**

**void main()**

**{**

**int arr[5]={2,3,4,5,6};**

**int i;**

**xyz(arr);**

**for(i=0;i<5;i++)**

**printf("%d ",arr[i]);**

**}**

**133.void main()**

**{**

**int A[]={2,4,6,8,10};**

**int B[]={3,6,9,12,15};**

**int C[]={4,8,12,16,20};**

**int \*ptr[3];//Array pointer**

**int \*\*pptr;**

**int i;**

**ptr[0]=A;**

**ptr[1]=B;**

**ptr[2]=C;**

**pptr=ptr;**

**for(i=1;i<=3;i++)**

**{**

**\*pptr+=i;**

**\*\*pptr+=i;**

**++pptr;**

**}**

**--pptr;**

**printf("\n%d\n",\*\*pptr);**

**for(i=0;i<3;i++)**

**printf("%d ",\*ptr[i]);**

**for(i=0;i<5;i++)**

**printf("\n%d %d %d",A[i],B[i],C[i]);**

**}**

**134.void main()**

**{**

**int a[-1];**

**a[-1]=23;**

**printf("%d",a[-1]);**

**}**

**135.void main()**

**{**

**int a[2]={2,3};**

**\*a=5;**

**\*(a+1)=40;**

**printf("\n %d %d",a[0],a[1]);**

**}**

**136.void main()**

**{**

**int a[2]={23,56};**

**a[2]=30;**

**a[3]=99;**

**printf("\n %d",\*(a+1));**

**}**

**137. main()**

**{**

**int a=511,b;**

**char \*ptr;**

**ptr=&a;**

**b=\*ptr;**

**printf("\n%d%d",a,b);**

**}**

**138. main()**

**{**

**int a=20;**

**int \*p=&p;**

**int \*\*p2p=&p;**

**++\*p;**

**++\*\*p2p;**

**a\*=a-2;**

**printf("\n%d",a);**

**}**

**139.void prints(char \*s)**

**{**

**puts(s);**

**if(\*s);**

**prints(s+1);**

**puts(s);**

**}**

**void main()**

**{**

**char s[]="colors";**

**prints(s);**

**}**

**140.void main()**

**{**

**char s1[4][40]={"cobol","pascal","java","oracle"};**

**char \*p[4],\*\*pp;**

**int i,j;**

**p[0]=s1[0];p[1]=s1[1];**

**p[2]=s1[2];p[3]=s1[3];**

**pp=p;**

**for(i=3;i>=0;i--)**

**{**

**\*pp+=i;**

**\*\*pp=i;**

**++p;**

**}**

**--p;**

**puts(\*pp);**

**for(j=0;j<4;j++)**

**puts(p[j]);**

**for(j=0;j<4;j++)**

**puts(s1[j]);**

**}**

**141.void main()**

**{**

**char s1[4][40]={"abc","123","xyz"};**

**char \*p[4]={"pascal","cobol"};**

**printf("\n %d",sizeof(p));**

**}**

**142. char \*change(char \*ptr)**

**{**

**++\*ptr;**

**++ptr;**

**ptr+=2;**

**\*ptr+=2;**

**return ptr-2;**

**}**

**void main()**

**{**

**char name[]="hello rahul";**

**puts(change(name));**

**puts(name);**

**}**

**143. void main()**

**{**

**char s1[40]="ABCDEFGH";**

**char \*s2="PQRSTRSV";**

**char \*s3="WXYZ";**

**s3=s2+2;**

**s2=s1+2;**

**++\*s2;**

**++\*s3;**

**puts(s1);**

**puts(s2);**

**puts(s3);**

**}**

**144.void main()**

**{**

**int a=10;**

**int \*p=&a;**

**int \*\*pp=&p ;**

**++\*p;**

**++\*\*pp;**

**a\*=a-2;**

**printf("\n%d",a);**

**}**